AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

A COLLECTION OF TECHNICAL PAPERS
ST. LOUIS, MISSOURI
JULY 22-24, 1985

**For permission to copy or republish,
contact the American Institute of Aeronautics and Astronautics,
1633 Broadway, New York, NY 10019**

A COLLECTION
OF
TECHNICAL PAPERS

AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

July 22-24, 1985/St. Louis, Missouri

Conference Committee

General Chairman
DOUGLAS W. LINDER
Northrop Corp.

Technical Program Co-Chairmen
FRANK CARDULLO
State Univ. of New York at Binghamton

MATT LANDRY
Bell Helicopter Textron

Administrative Chairman
LARRY ROSS
McDonnell Douglas Corp.

TABLE OF CONTENTS

AIAA NO.		PAGE NO.
85-1730	Application of the Dynamic Flight Simulator (DFS) to Evaluate Pilot Performance in a Simulated F-14 Flat Spin Environment - J. EYTH, JR. and D.P. GLEISNER	1
1732	Simulator Evaluation of F/A-18 Ski Jump - B.L. DOUGHERTY and D.R. ROLSTON	6
1734	AV-8B Harrier II Training Capabilities - R.J. MUFFLER	11
1738	Simulating the Works: From Crew Input to System Response - R.A. WEEKS	16
1740	Optimal Generalized Multistep Integration Formulae for Real-Time Digital Simulation - D.D. MOERDER and N. HALYO	22
1741	Flight Simulation Fidelity in a Total G-Force Environment - D.E. BISCHOFF, D.A. POOLE and J. EYTH, JR.	29
1742	Controlling the Human Centrifuge as a Force and Motion Platform for the Dynamic Flight Simulator - R.J. CROSBIE and D.A. KIEFER	37
1743	Using Human Motion Perception Models to Optimize Flight Simulator Motion Algorithms - K.S. FORSSTROM, J. DOTY and F.M. CARDULLO	46
1748	A Simulated Infrared Model Board - C.E. JONES and J. LEE	52
1749	A Ring-Vortex Downburst Model for Real-Time Flight Simulation of Severe Wind Shears - M. IVAN	57
1750	Composite Statistical Method for Modeling Wind Gusts for Aircraft Simulation - J.R. SCHIESS	62
1752	A Flexible Design for a General Purpose Simulation Facility - A.C. CRUCE and S.A. ALEXANDER	69
1754	A Real Time Executive for the Control of a Multicomputer Simulation Complex - J.H. CROFT	81
1756	Aircraft Tactical Environment Simulation for the 1990's - S. RAMACHANDRAN, D.K. OLDHAM and R.L. BRANSON	92

LATE PAPER

85-1739	Rotor Dynamics Simulation: A New Approach - F.M. CARDULLO, W.J. HEWITT and K. KNIGHT	97

85-1730

APPLICATION OF THE DYNAMIC FLIGHT SIMULATOR (DFS) TO EVALUATE PILOT PERFORMANCE IN A SIMULATED F-14 FLAT SPIN ENVIRONMENT

Jacob Eyth, Jr.* and LT. David P. Gleisner, USNR**
Naval Air Development Center, Code 6022
Warminster, PA 18974-5000

Introduction

This paper details the preparation, conduct and results of a NAVAIRSYSCOM funded program to investigate the aircrew safety problems associated with the steady state flat spin mode of the F-14A aircraft. The simulator utilized for the testing was the Dynamic Flight Simulator (DFS) located at the Naval Air Development Center, Warminster, PA. The DFS is a unique flight simulation device which uses the Center's 50-foot arm human centrifuge as a motion base. The DFS is the only ground-based full-capability flight simulator able to reproduce the multidirectional sustained G environment of actual flight. With this capability researchers were able to safely analyze the effects of the sustained eyeballs-out G-environment of the F-14 flat spin.

Like many high performance aircraft the F-14 exhibits several spin modes which can develop if the plane departs from controlled flight. In certain cases, without timely recovery control inputs, the aircraft may transition into an aerodynamically stable flat spin mode (90° AOA with increasing yaw rate to 180°/sec). Based on the distance of the pilot from the perpendicular spin axis of the aircraft (23 feet) the pilot can be subjected to eyeballs-out G-forces as high as 6-1/2 G's. Pilots who have experienced the eyeballs-out G-forces during a flat spin have indicated almost total incapacitation if their shoulder harness is not securely locked.

Results of the NAVAIRDEVCEN flat spin tests uncovered several previously unknown aspects of the pilot's capabilities in this environment. Two proposed spin warning displays and an automatic locking restraint system, intended to aid the pilot in both the recognition of and recovery from the simulated spin, were also evaluated.

System Configuration for the Tests

The Dynamic Flight Simulator (DFS) is a high fidelity flight simulator which can be operated independently or as an integral part of the NADC Human Centrifuge. (Figure 1). A block diagram of the overall DFS system is shown in Figure 2. Functionally, the simulator can be broken down into four primary components: (1) a multi-purpose crew station containing an F-14A aircraft cockpit, (Figure 3); (2) the human centrifuge motion platform; (3) a simulation control area linking the crew station and centrifuge to a series of computers which manipulate and transfer real-time digital and analog data; and (4) the NADC Central Computer System which is used to perform the system's aerodynamic modeling and data processing.[1]

Human Centrifuge Motion Base

The NADC human centrifuge is the largest three degree-of-freedom, centrifuge in the world. Its unique features include: a 50-foot long arm which minimizes G gradient and Coriolis force problems; a 10-foot diameter gondola suspended in a controllable dual gimbal system; a 16,000 hp drive motor capable of providing an average 10 G/second onset rate between 1.5 and 15G; and a 40,000 G-pound payload capability. During the spin exercises the G-limits were set at +5G's in the vertical axis (G_z), -5G's in the longitudinal axis (G_x), and ±1g in the lateral axis (G_y). G-onset rates were limited to 2-1/2 G's/sec.

The successful utilization of the human centrifuge as the motion base for the DFS required the implementation of a new centrifuge control algorithm.[2] The algorithm improves the fidelity of the pilot's perceived angular motions by masking angular artifacts with linear accelerations. The result is a realistic sensation of flight without contradictory motion cues.

F-14 Aerodynamic Model

The simulation math model incorporated in the DFS is the same F-14 model used by NASA-Dryden, NAVAIRTESTCEN and Grumman Aerospace Corporation.[3] The algorithms encompass the full nonlinear, 6 degree of freedom, rigid body equations of motion of the F-14A aircraft. They are designed to accommodate large angle orientations; mass, engine and store asymmetries; and classical, as well as rotary balance, stability and control derivatives. The specific F-14 implementation used in this study is applicable to a clean (no stores) loading in the cruise (landing gear and flaps up) configuration with automatic maneuvering flaps and slats.

System Aerodynamic Validation

Prior to beginning the F-14 Spin Program, the DFS underwent extensive validation using qualified NAVAIRDEVCEN flight test engineers and Navy and contractor test pilots. The validation testing confirmed the accuracy of the F-14 model in the batch mode, fixed base real-time mode, and dynamic mode (with the centrifuge operational).

The test pilot evaluation determined that visual and motion cues were most realistic in the post stall gyration, departure and steady state spin regimes. Subsequent evaluations by fleet pilots during the Flat Spin exercise confirmed this observation.

Spin Recovery Test Program

Test Objective

The primary purpose of the Spin Recovery experiment was to determine from which, if any, yaw rates F-14 pilots would be able to recover the simulated air-

* DFS Program Manager, Member AIAA

** Aerospace Experimental Psychologist, Member AIAA craft after it had entered a flat spin. The exercise was also expected to uncover successful from unsuccessful recovery control inputs, and whether unsuccessful recoveries are primarily a problem of mental confusion or the physical inability of the pilot to perform the correct control movements.[4]

Independent Variables

The objectives of the test program were addressed by having pilots perform spins at various yaw rates of up to 155 deg/sec (-5Gx), then assessing the effectiveness of control inputs by measuring time elapsed and altitude lost until the aircraft was brought under control.

In addition to recovery performance measurements, improvements offered by a -Gx automatic-locking restraint system and two spin warning displays were evaluated.[5] The restraint system tested allows the pilot freedom of movement until the aircraft senses -0.8Gx, at which point the harness automatically locks. This feature allows the pilot to concentrate on the correct recovery control inputs without having to remember to lock his harness while trying to remain in position and control the aircraft. To compare the benefits of the auto-locking restraint system, pilots were asked to perform one spin to -3Gx in a completely unrestrained condition.

The first of the spin warning displays tested was a spin directional arrow and yaw rate indicator (ECP-Tape 113A). This display (Figure 4) is available to the pilot on the tactical information display (TID) repeat-mode of the horizontal situation display (HSD). It consists of an arrow pointing in the direction of the spin with a continuous readout of the yaw rate beneath it. The display appears on the HSD when the aircraft reaches a 30 deg/sec yaw rate and starts to flash at 90 deg/sec. During recovery, as the yaw rate decreases below 30 deg/sec, the spin arrow disappears and the HSD returns to the previous display format.

The second display (Figure 5) was a stick position indicator. It consisted of a pictorial display showing the actual position of the stick in relation to the total area of stick movement. This display was designed based on the theory that an out-of-position pilot struggling against acceleration forces would benefit from knowing where the stick actually is in relation to where he is trying to position it for spin recovery.

Pilot/Subjects

Seventeen pilots participated in the data collection phase of the study. Eleven of the pilots were current F-14 fleet pilots with at least 400 hours in the F-14; two of them were T-2 spin instructors; and four others were test pilots from NAVAIRTESTCEN, NASA-Dryden and Grumman Aerospace Corporation.

Each pilot received approximately 20-30 minutes of static practice and 15 minutes of dynamic practice before any data was collected. The practice consisted of typical flight maneuvers such as 30^{o} and 60^{o} roll reversals, 1G barrel rolls, pitch doublets, 4G steady banked turns, split S with aileron rolls, inside loops and low G entry spins. For all dynamic runs the pilots wore full flight gear including an anti-G suit, survival vest, torso harness, helmet, and oxygen mask with the hose removed.

Each data collection session started with a static "warm-up" spin which served as a system check and also to get the pilot in the proper frame of mind. Data was collected on 3 to 6 spins during each dynamic session with the number depending on the amount of pilot fatigue. Pilots participated in 1-3 sessions depending on their availability.

To enter the spin, the pilots were instructed to perform a wings level deceleration to 120 knots at 30,000 feet altitude, apply full pro-spin cross-controls (e.g., aft right stick and left rudder) and hold them until the desired yaw rate was reached at which point they were told to recover the aircraft according to NATOPS[6] procedures. Most spin recoveries began at an altitude of 20,000-22,000 feet. Total elapsed time for each spin maneuver was 2-3 minutes from the time that pro-spin controls were applied. The pilots were allowed as much rest time between spins as they desired.

Test Results

Quantitative Results

In all, this study yielded quantifiable results on eighty-eight (88) spin simulations as flown by the seventeen pilot subjects. Tabular data was obtained which correlated number of revolutions, altitude loss and recovery time as functions of the entry spin conditions. These data were manually extracted from strip chart recordings taken during each run. Spin recovery was defined as complete when yaw rate returned to zero and the aircraft was back in controlled flight. An additional 3,000-4,000 feet would be required to return the aircraft to level flight from the nose-down condition at the end of the spin recovery.

Revolutions to Recovery

The effect of entry yaw rate upon the number of revolutions experienced before recovery is graphically summarized in Figure 6. As expected, the number of revolutions increases as a positive function of the yaw rate present at the time recovery is initiated. Pilot activation of Roll SAS during the early stages of spin recovery resulted in a trend toward reduced revolutions before recovery. At an entry yaw rate of 120^{o}/second ($-3G_x$), it took an average of 5.5 revolutions to recover with SAS ON. Without SAS, the average revolutions preceding recovery from this rate was just over 7.5 turns. If yaw rate is allowed to build to 139^{o}/second ($-4G_x$) before recovery is initiated, the average revolutions to recovery increases to just over 7.0 for SAS ON and to 9.0 when SAS is not used. When the yaw rate is allowed to reach 155^{o}/second ($-5G_x$) before recovery is attempted, the average number of turns was 10.0 with SAS ON and 17.0 revolutions with SAS OFF.

Altitude Loss to Recovery

The development of higher yaw rates also led to a greater altitude loss during recovery. (See Figure 7). As with revolutions to recover, there is a trend for altitude loss to be greater when the roll SAS is OFF. Under the least severe condition (120^{o} yaw rate, $-3G_x$), the average altitude loss experienced was just over 5,000 feet with SAS ON and 7,000 feet with SAS OFF. When the yaw rate was allowed to increase to 139^{o}/second ($-4.0\ G_x$) before recovery was attempted, the impact of Roll SAS activation on altitude loss increased markedly with an average loss of 6,400 feet with SAS ON and compared to a loss of 7,300 feet when the SAS was not activated. At 155^{o}/sec (-5G$_{x}$) altitude loss was 7,000 feet with SAS ON and 12,500 with SAS OFF.

Effects of Altitude at Spin Entry

A cursory examination of the effect of altitude on simulated spin recovery characteristics was conducted by performing spins from three trim altitudes: 38,000; 30,000; and 20,000 feet. The aft longitudinal stick recovery technique with Roll SAS OFF was utilized in this investigation. The resultant altitude loss, time and number of turns from the initiation of recovery controls to zero yaw rate are presented in Figure 8. Although insufficient data were obtained from which to draw definitive conclusions, the trend was toward reduced altitude loss, time, and number of turns to recover as the recovery initiation altitude was lowered. The primary effect was in altitude loss. At the higher entry altitude (35,000 feet), the altitude loss per turn was 1,360 feet. At lower altitudes (20,000 feet), the altitude loss was 1,140 feet per turn due to increased air density.

Displays

The type of display provided to the pilot did not have a quantifiable impact on recovery performance as measured by the average number of revolutions or altitude loss. Neither the spin arrow (Figure 4) or the stick display (Figure 5) were found to reliably enhance or degrade recovery performance. However, these displays did elicit a number of subjective comments from the pilots. The spin arrow/yaw rate display was felt to be most helpful as an indication of increasing or decreasing trends. Pilots were better able to assess their chances for recovery with a yaw rate read-out. The stick display was found to be of marginal value unless its function was enhanced to provide instructional information on where the stick should be placed for optimum recovery.

Restraint System

The Flat Spin study provided an excellent opportunity to evaluate the utility of the NADC advanced development automatic locking restraint system. Since, in the interest of pilot safety, no spin exposures were programmed beyond 120^{o}/second (-3G$_{x}$) with an unlocked restraint system, there was no opportunity, to compare directly, the benefits of either the pre-locked or auto-locked restraint systems to an unrestrained condition at the higher -G$_{x}$ conditions where the advantages would be even more apparent. However, the subjective comments of the pilots who evaluated the auto-locking feature were unanimous in their praise of the system and recommended its immediate introduction into fleet aircraft.

Discussion

Overall results of the testing must be tempered by the fact that the DFS produces a simulated F-14 flat spin, a condition which has not been successfully flight demonstrated. The nature of the test plan required that the pilot deliberately fly the aircraft into a flat spin, something he would never do in actual flight. Consequently, the psychological aspects of surprise, confusion, and a life-threatening situation were missing. Because most spin entries occurred at 30,000 feet in altitude, the pilots had sufficiently recovered the aircraft before 10,000 feet, the altitude at which they are trained to eject. Had the spins been entered at 20,000 feet the number of successful recoveries would have been reduced.

The most significant outcome of the test was that pilots were not physically incapacitated by the high -Gx forces. Although pilots tended to differ widely in their body size and arm strength, no pilot had problems applying aft-stick and full rudder inputs under the eyeballs-out g-forces which were produced during the tests. (-5 Gx restrained, and -3 Gx unrestrained).

A secondary outcome of the tests was the successful utilization of the DFS as an out-of-control flight environment testing device. The fully developed flat spin simulation which can be done in the DFS is a flight regime which cannot be tested safely in flight tests. The total accumulated data on F-14 flat spin accidents amounts to approximately 20 incidences in the Navy's Safety Center files. DFS pilots as a group have flown over 100 simulated F-14 spins.

The results of this study have demonstrated that development and refinement of the DFS as a research and development tool is warranted. The DFS is unique in its crew system testing capabilities and its potential for developing new systems which may save pilots and aircraft. With a better understanding of human problems facing pilots in severe flight situations, crew system developers in government and industry should be able to respond more quickly and effectively to the needs of the fleet.

References

1. Crosbie, R. J. and Eyth, J., "A Total G-Force Environment Dynamic Flight Simulator," AIAA Paper No. 83-0139, presented at The 21st Aerospace Sciences Meeting, Reno, Nevada, 10-13 January 1983.

2. Crosbie, R., "Application of Experimentally Derived Pilot Perceptual Angular Response Transfer Functions," Proceedings from the AIAA Flight Simulation Technologies Conference and Technical Display, Niagara Falls, NY, 13-15 June 1983.

3. Stifel, J.M., Math Model for NADC Spin Simulator and F-14 High Angle of Attack Data Base, Volume 1. Report No. NADC-80220-60, Naval Air Development Center, Warminster, PA, October, 1980.

4. Hitchcock, L. Proposed Protocol for the Initial DFS F-14 Spin Study. Technical Note 51216-02, Essex Corporation, Warminster, PA, February 1983.

5. Bateman, R.P., Biers, D.W., Ramirez, S.D. Dynamic Flight Simulator F-14 Spin Simulation Program, Volumes I and II. Report No. NADC-81285-60, Naval Air Development Center, Warminster, PA, June 1982.

6. F-14A NATOPS Flight Manual, NAVAIR 01-F14AAA-1, December 1982 with current changes.

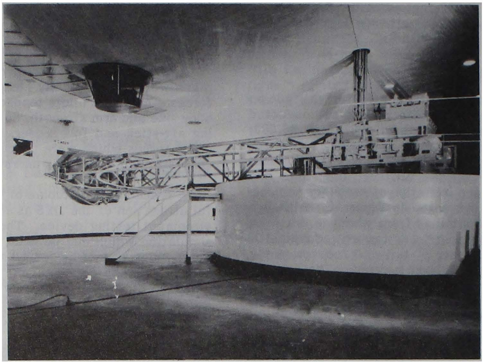


FIGURE 1. NAVAIRDEVCEN HUMAN CENTRIFUGE

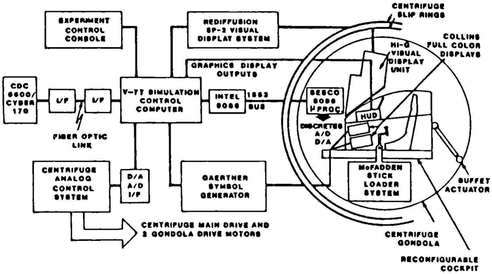

EXPERIMENT CONTROL CONSOLE
REDIFFUSION SP-2 VISUAL DISPLAY SYSTEM
CENTRIFUGE SLIP RINGS
COLLINS FULL COLOR DISPLAYS
HI-G VISUAL DISPLAY UNIT
GRAPHICS DISPLAY OUTPUTS
CDC 6600/ CYBER 170
I/F
I/F
V-77 SIMULATION CONTROL COMPUTER
INTEL 8086
1553 BUS
BESCO 8086 μPROC
HUD
DISCRETES A/D D/A
FIBER OPTIC LINK
CENTRIFUGE ANALOG CONTROL SYSTEM
D/A A/D I/F
GAERTNER SYMBOL GENERATOR
McFADDEN STICK LOADER SYSTEM
BUFFET ACTUATOR
CENTRIFUGE MAIN DRIVE AND 2 GONDOLA DRIVE MOTORS
CENTRIFUGE GONDOLA
RECONFIGURABLE COCKPIT

FIGURE 2. DYNAMIC FLIGHT SIMULATOR (DFS) BLOCK DIAGRAM

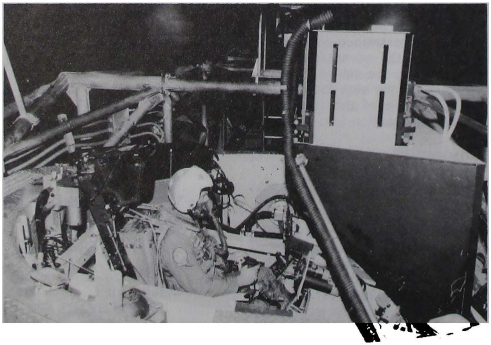


FIGURE 3. DFS F-14A CREW STATION

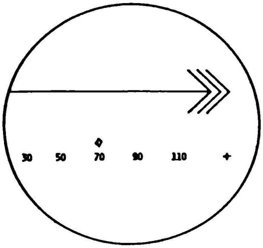


FIGURE 4. SPIN ARROW DISPLAY

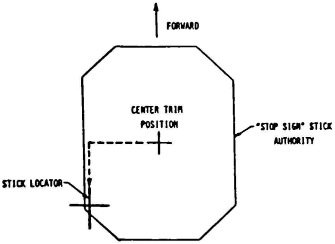


FIGURE 5. STICK POSITION DISPLAY

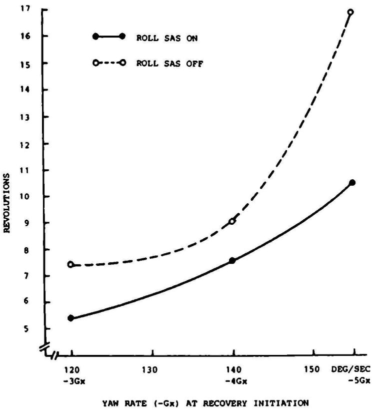


FIGURE 6. REVOLUTIONS TO RECOVERY vs YAW RATE

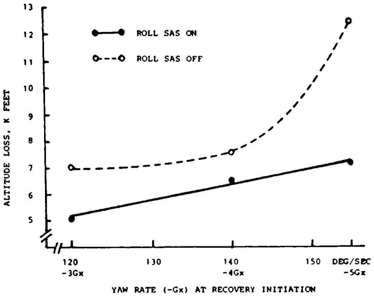


FIGURE 7. ALTITUDE LOSS TO RECOVERY vs YAW RATE

ALTITUDE AT RECOVERY INITIATION (1) (FT)	ALTITUDE LOSS (FT)	# TURNS	TIME (SEC)
35,000	10,200	7.5	33
26,000	8,600	7.2	32
20,000	7,400	6.5	30.5

(1) G_X=3.35g (130 deg/sec yaw rate)

FIGURE 8.
VARIATION OF RECOVERY PARAMETERS
AT DIFFERENT SPIN ENTRY ALTITUDES

85-1732

SIMULATOR EVALUATION OF F/A-18 SKI JUMP

B. L. Dougherty*
D. R. Rolston**
McDonnell Aircraft Company
St. Louis, Missouri

Abstract

The paper discusses the development and evaluation of the simulator effort involved in a recent Navy ski jump program. It was found that update rate and detail design of the landing gear and ramp models were critical factors in producing this highly effective man-in-the-loop simulation. Comparison of simulator and flight test data demonstrate the usefulness of this simulator in augmenting the flight test program.

Nomenclature

F_{g_x}	x-axis gear reaction force component
F_{g_x}	z-axis gear reaction force component
h	ramp height
R	radius of curature
Δt	sample interval time increment
V	aircraft velocity
x	horizontal position along ramp
z	vertical position on ramp
θ	pitch rate

Introduction

The Naval Air Test Center requested preliminary analysis of a ski jump assisted take-off for the F/A-18 Hornet in mid-1981. Concurrently, tests and analyses were planned for both the T-2C trainer aircraft and the F-14A fighter. These three aircraft represent the progression of flight control system designs in the last 10 to 15 years from the mechanical T-2C to the digital fly-by-wire F/A-18.

The stated purposes of these evaluations were to (1) test the feasibility of the ski jump concept, (2) define the operational limitations, (3) document performance gains, (4) propose design considerations and (5) verify the aerodynamic and structural ski jump analysis program.

A. Program Components - The F/A-18 simulation effort was comprised of three parts, (1) a computerized structural analysis of the nose and main landing gear; (2) a six-degree-of-freedom analysis of the aerodynamic performance gains; and (3) the manned simulator effort to integrate the aerodynamics, flight controls, engines, landing gear and avoinics in a real-time man-in-the-loop environment.

The first two components are extensions of the basic airframe and aerodynamics evaluation programs and required only an accurate ramp model to produce performance predictions. With the absence of a pilot-in-the-loop requirement, simplifications can be made in flight control and engine modeling. In these simulations an extremely small computation sample interval can be used thus simplifying the digital integration techniques. The manned simulator must update at a rate which is consistent with both data accuracy and pilot perception. The methods of integration, system order and update rate must all be optimized to provide the pilot with an accurate representation of the "total" aircraft. The major task, therefore, is to analyze these components and evaluate the effects and trade-offs to produce an acceptable real-time simulation.

B. Simulator Facilities - The real-time flight simulation effort was conducted at the McDonnell Aircraft Manned Air Combat Simulator (MACS) facility in St. Louis. The F/A-18 Flight Hardware Integration Cockpit (MACS 3.5) was selected for use because of its precise flight control and aerodynamics modeling. This simulator is a fixed base, all digital system and provides the pilot with full cockpit instrumentation and production cockpit layout. It is used primarily for flight control and handling qualities evaluation and development. It is equipped with flight hardware displays and mission computers which permits a larger portion of the host computer memory and computation time to be available for the essential handling qualities models. A Vital IV visual system is installed to give the pilot an out-the-window display of the runway, ramp, and airfield environment. As a flight hardware integration facility, MACS 3.5 is equipped with full displays and Heads-Up Display (HUD) identical to the F/A-18 Hornet used in the flight phase of the program.

C. Simulator Evaluation Program - The manned simulator evaluation program was conducted in two parts. The first portion included model verification, pilot familiarization, preliminary analysis of take-off roll, trim settings and thrust and flap setting effects. This session consisted of several hundred ski jump take-offs using 3, 6 and 9 degree ramps. Engineers from Naval Air Test Center and Naval Air Development Center used this data to develop the initial flight test program plan.

The second effort, in mid-1983, was used to determine minimum acceptable end speeds, required ground roll and trim settings for the two gross weights to be used in flight testing. Additionally, effects of cross winds and engine failures were evaluated. Pilot reaction time and aircraft dynamics were examined to determine abort criteria and ejection envelopes. The coordinated simulation effort was designed to develop a test plan and test procedures to best evaluate the ski jump take-off.

* Senior Engineer - Laboratory
Member AIAA

** Chief Laboratory Engineer

During these manned simulator evaluations, the Navy pilots developed a pitch attitude capture technique using the flight hardware Heads-Up Display in the simulator. This technique put the pilot in the loop with the aircraft to hold a 15 degree pitch attitude off the ramp. This produced additional performance gains and was included in the test plan for the flight evaluation.

Background

A brief discussion of the ski jump ramp dynamics is presented as introduction to the performance gains anticipated by this project.

A. The Ski Jump Ramp - The ski ramp provides a method of vectoring the flight path forward velocity component to augment its vertical velocity component. This technique permits an aircraft to rapidly attain altitude which can subsequently be exchanged for airspeed. It provides an abrupt rotation and departure from the take-off surface, which helps overcome aerodynamic and ground effects resisting acceleration and rotation for lift-off during conventional take-offs. The benefits of this technique are (1) to reduce required flying speed at lift-off and (2) to increase the take off gross weight for a fixed runway length.

The following example illustrates the ski jump dynamics using a single fixed radius of curvature and a point mass aircraft. The ramp height is defined as:

$$h = \frac{X^2}{2R} \qquad (1)$$

where R is the ramp radius of curvature and X is the horizontal position on the ramp. The ramp departure angle determines the aircraft pitch angle (θ) at the end of the ramp, this angle is defined as:

$$\theta = \arcsin \frac{X}{R} \simeq \frac{X}{R} \text{ for small angles} \qquad (2)$$

The ramp end speed (V) is the vector sum of the horizontal and vertical velocities. For small ramp angles the end speed is approximately the horizontal velocity ($^{dx}/_{dt}$).

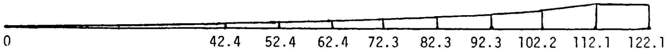
The key analysis parameters in predicting performance gains are the vertical velocity ($\dot{h}$), pitch rate ($\dot{\theta}$) and the vertical acceleration ($\ddot{h}$). For a rigid body aircraft following this ramp, these parameters are derived to be:

$$\dot{h} = \frac{XV}{R} \qquad (3)$$

$$\dot{\theta} = \frac{V}{R} \qquad (4)$$

$$\ddot{h} = \frac{V^2}{R} \qquad (5)$$

The assumptions of fixed radius, rigid body and small angles are all valid for the initial predictions. However, the effects of aerodynamic effects and landing gear dynamics must be included to accurately predict performance gains. The effect of the flight control system must be considered to determine the fly-out characteristics. However, approximate verification of the ramp model can be done by comparing simulator vertical velocity results with the simplified prediction.



DISTANCE ALONG RAMP (ft)	RAMP HEIGHT (ft)		DISTANCE ALONG RAMP (ft)	RAMP HEIGHT (ft)	
	6 deg	9 deg		6 deg	9 deg
0	0	0	82.3	3.88	4.40
42.4	1.16	1.16	92.3	4.81	5.62
52.4	1.68	1.71	102.2	5.85	7.02
62.4	2.30	2.44	112.1	5.85	8.58
72.3	3.03	3.33	122.1	-	8.58

Figure 1

B. Flight Test Ramp Geometry - The diagram in Figure 1 shows the geometry of the existing ramp at Patuxent River. The simulator ramp model was developed to replicate this configuration. During this evaluation, ramp angles of 3, 6 and 9 degrees were used.

Discussion

A. Ramp Modeling - In modeling the ramp, two approaches were considered. The first used the equation for the ramp to determine the elevation of the ramp at any distance along the ramp. This method makes maximum use of the computational power of the digital computer. The second method used a table look-up format where the distance along the ramp is input to determine the local ramp height. This method requires storing the ramp height data at given break points and using linear interpolation between break points. This second method provides an accurate representation of the actual ramp at NATC, which is constructed of flat sections.

B. Landing Gear Dynamics Model - The landing gear forces and moments were computed using a model developed from two simulation programs performed by McDonnell Aircraft for the NASA Langley Research Center (Contracts NAS 1-23378 and NAS 1-13981, "Expansion of Flight Simulator Capability for Study and Solution of Aircraft Directional Control Problems on Runways"). This model has been extensively tested and found to faithfully represent the landing gear performance of the F/A-18.

The relative geometry between the strut stroke line for each landing gear and the runway surface was detailed so that each strut stroke and the corresponding stroke rate can be computed. The spring and damping forces are computed from the stroke and stroke rate and data describing the strut load stroke and damping characteristics. The model uses the relative geometry data is used to compute the ground velocities, tire skid angles and the skid and drag coefficients. From these coefficients and the strut ground reaction force, the skid and drag forces are computed at each tire contact point. These forces are resolved into body axes, and moments about the aircraft center of gravity are computed.

C. Error Sources - In development of the ski jump simulation, two potential error sources were identified, update rate and landing gear model accuracy. The update rate affects the integration techniques available and the frequency bandwidth of the dynamic models. The landing gear forces and moments produced by the model must be realistic in both magnitude and time history to provide the appropriate response to the ramp.

In Figure 2, the main fly-out parameters are plotted for two identical take-offs using different update rates. These cases used a short (42 foot) ramp with an end speed of 144 knots. The shorter update interval (.01) seconds showed a lower peak pitch rate and a higher peak vertical velocity. However, the angle of attack curves are identical. The simplified ramp analysis yields a peak vertical velocity of 10.2

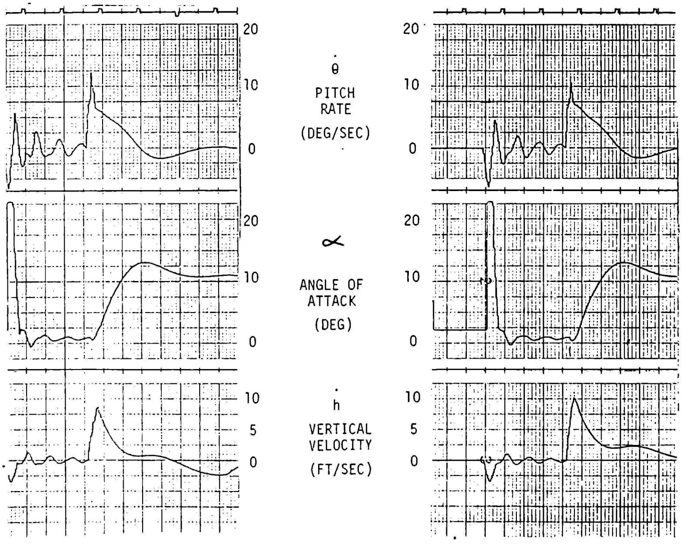


Figure 2

feet per second. Therefore, it can be seen that the .01 second update rate produces a result much closer to the analytical result than the .03 second update rate. The primary reason for this is the increase in the number of passes made through the landing gear model. At the .03 update rate the simulation computes 6 passes on the ramp, while the .01 second rate computes 16 - 17 passes on the ramp. Updating the entire simulation at the .01 second rate proved to be unfeasible. Therefore a looping technique was used to update the landing gear, aerodynamics and equations of motion at the higher rate while computing the remainder of the simulation at the slower rate.

The landing gear model computes the current forces and moments based on the changes in strut compression between passes. The more times these calculations are updated, the more precise the results. The strut reactions must also account for the rise of the ramp. The angle of the ramp also effects the direction of the gear reaction forces.

The equations of motion use the current body position, rates and accelerations to determine aircraft state at the end of the subsequent cycle using a predictor method. Therefore, these equations do not include the effects of the rigid ramp on the subsequent update cycle. The actual aircraft reaction forces must be recomputed in the landing gear model to account for the change in ramp height. The correction must account for both the difference in aircraft height due to the ramp and the change in local ramp angle and its effect on gear reaction forces. The form of this correction is:

$$X_{i+1} = X_i + U_X \Delta t \tag{6}$$

$$Z_i = f(X_i) \tag{7}$$

$$Z_{i+1} = f(X_{i+1}) \tag{8}$$

$$\theta_{1_{i+1}} = (Z_{i+1} - Z_i) / (X_{i+1} - X_i) \tag{9}$$

$$X'_{i+1} = X_{i+1} - (Z_{i+1} - Z_i)\, \theta_{1_{i+1}} \tag{10}$$

$$Z'_{i+1} = f(X'_{i+1}) \tag{11}$$

$$\theta'_{1_{i+1}} = (Z'_{i+1} - Z_i) / (X'_{i+1} - X_i) \tag{12}$$

Where X is aircraft X location at time i, Z is ramp height at time i and θ is the current (time i+1) local ramp angle. This correction is applied one or more times to improve the accuracy of the corrected parameters. A single correction was found to improve accuracy significantly, but multiple corrections provided little additional accuracy. The gear reaction forces are the corrected by the equations:

$$F'_{g_z} = F_{g_z} \cos \theta_{1_{i+1}} \tag{13}$$

$$F'_{g_x} = F_{g_2} \sin \theta_{1_{i+1}} + F_{g_x} \tag{14}$$

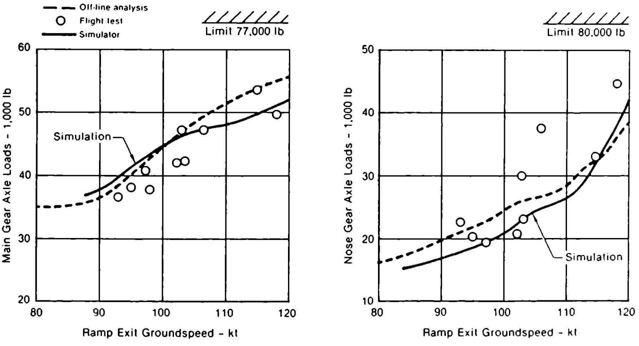


Figure 3
F/A-18A Structural Loads 37,000 Lb Gross Weight
9 Deg Ski Jump Ramp

Results

The true measure of the simulator's effectiveness in predicting performance is validation by flight testing. The flight test data from NATC is compared below with the results of the simulator evaluation.

A. Landing Gear - The structural loads expected in both the main and nose landing gear were investigated through off-line gear model analysis. These results determined the limitations placed on the test plan due to structural considerations. During both simulator and flight evaluations, landing gear loads were recorded to verify these results and to correlate simulator and flight test data.

Figure 3 plots the peak structural loads for a 37,000 lb. gross weight F/A-18 on the 9 degree ramp. The off-line and simulator loads tend to bracket the flight test results. The higher nose gear loads observed in flight testing were attributed to NATC to the uneven matting material used in the ski jump ramp approach path. This matting tended to cause an additional stroking of the nose gear at higher ground speeds.

B. Flyout Performance - Between September 1983 and February 1984, NATC conducted 91 ski jump take-offs with the F/A-18. Figure 4 shows the minimum vertical velocity attained on the 6 degree ramp for both flight test and simulator. The flight test data varied slightly from the simulator. However, it should be noted that the predicted minimum end speed and that actually attained corresponded within approximately 1 knot. The performance improvement attained by the pilot's pitch attitude capture technique is clearly discernable. The normal field take-off speed for the 37,000 LB aircraft is 154 knots. The 6 degree ramp reduces this to approximately 107 knots. The pitch capture technique further reduces this to 98 knots.

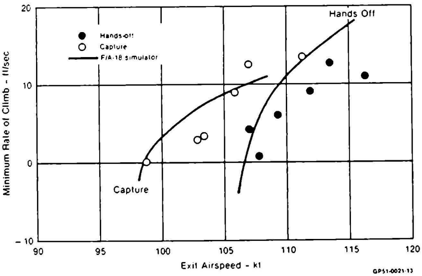


Figure 4
Minimum Rate of Climb on 6° Ramp

The highest ground speed used in the flight test program was 125 knots. This is well below the normal ground speed for take-off. The extensive use of the simulator allowed for this significant reduction to be performed with confidence. This also greatly reduced the number of take-offs required to safely reach the minimum end speed.

Conclusions

The F/A-18 ski jump program benefited significantly from the extensive simulator effort. The simulator provided model validation for off-line programs, as well as pilot familarization and training. It established safe operation envelopes and made a significant contribution to test plan development.

The manned simulator provides the link between off-line parametric studies and flight test program. It can be used to reduce the number of aircraft configuration, trim conditions and power setting used in flight test. The simulator can also be used to evaluate pilot technique, procedural considerations and safety factors.

The manned flight simulator of high fidelity provides a relatively inexpensive tool that greatly enhances the effectiveness of a flight test program. NATC reported that "simulation is the perfect tool to evaluate different ski jump profiles..."

Bibliography

1. Clark, J. W. Jr., CTOL Ski Jump Dynamics Analysis Model and Computer Program. Final Report, NATC, June 1983.

2. Haag, E. A., F/A-18A Predicted Ski Jump Takeoff Characteristics, Vol. 1 - F/A-18A Ski Jump Structural Loads, McDonnell Douglas Corp., Nov. 1982.

3. Pulley, C. M., F/A-18A Predicted Ski Jump Takeoff Characteristics, Vol. 2 - Performance and Flying Qualities, McDonnell Douglas Corp., Nov. 1982.

4. Rolston, D. R., Sinusoidal Integration for Simulation of Second-Order Systems, AIAA Simulation Technologies Conference - 1983, June 1983, pp 52-63.

5. Senn, C. P., Eastman, J. A. and Wagner, T. A., Conventional Takeoff and Landing (CTOL) Airplane Ski Jump Evaluation. 14th Annual Symposium Proceedings, 1983, Session 3 - Test Results, Society of Flight Test Engineers, Aug. 1983.

85-1734

AV-8B HARRIER II TRAINING CAPABILITIES

R. J. MUFFLER, UNIT CHIEF
FLIGHT SIMULATION
MCDONNELL AIRCRAFT COMPANY
MCDONNELL DOUGLAS CORPORATION

INTRODUCTION

In any aircraft program the amount of actual flight time available to the individual pilot is limited. The McDonnell Douglas AV-8B program is no exception. Simulation cannot replace actual flight time, but it can significantly improve the efficiency of actual flight time by increasing the quality and level of proficiency attained by the individual aviator.

The need for training large numbers of replacement pilots for the Marine Corps AV-8B requires a device with a high level of fidelity in all areas of the aircraft flight envelope. A single device with these capabilities could be accomplished but would unnecessarily limit the rate of pilot training. Since pilot training is accomplished in an incremental manner, a suite of trainers were developed which would provide complementary capability fulfilling the total training needs while allowing multiple training missions simultaneously.

Each of these trainers will provide training in a different area of aircraft performance. For the Marine Corps three trainers will be available: 1) The Aircraft Systems Trainer (AST) 2) The Operational Flight Trainer (OFT) and 3) The Weapons Tactics Trainer (WTT). It is the latter two that will be the discussed in this paper since it is those trainers that advance the state-of-the-art in aircraft simulation for fighter/attack missions.

OFT TRAINING REQUIREMENTS

The OFT will have as its primary mission the training of pilots in those unique skills and techniques associated with Vertical/Short Take-off and Landings (V/STOL). In addition it will train basic aircraft control, instrument flight procedures and normal/emergency mode operations for AV-8B aircraft systems. The V/STOL simulation includes vertical take-off/land, ski-jump take-off, rolling vertical take-off/land, short take-off/land and conventional take-off/land as well as all acceleration/deacceleration transition characteristics. The OFT has the capability for enroute navigation using TACAN radio navigation, inertial navigation system, or visual navigation across a specific real world data base. In addition, all cockpit procedure training including Pre/Post Start Checklist, Engine Start, Pre-Taxi, Taxi and Pre-Take-Off Checklists are a required trainer capabilities.

All of these training tasks are to be accomplished in a wide variety of environments including land and water with surrounding air space. Specific airfields, landing platforms, carriers, amphibious assault ships and remote confined area landing sites will be a part of the training requirements including the capability to control time of day in the environment for both day and night time simulation.

Since the primary goal is to maximize the effectiveness of flight time the simulation fidelity must be extremely accurate to avoid any re-learning when transitioning from simulator to aircraft and vice versa. It is this philosophy combined with the training tasks outlined that has driven the design to the highest degree of realism and accuracy achieved in any fighter/attack trainer to date. The special capabilities of individual simulation systems will be discussed under the section on design concepts.

WTT TRAINING REQUIREMENTS

The WTT has a role, complementary to the OFT, of training pilots in Air-to-Ground and Air-to-Air Weapons Delivery, Defensive Electronic Counter Measures as well as normal/degraded mode operations for all aircraft systems. Training may be accomplished in areas such as low altitude navigation and target penetration with weapons delivery for all types of ordnance and all modes of attack. These weapon deliveries are then scored at the instructor station for evaluation of student weapon delivery missions.

Additional training requirements of the WTT include offensive Air Combat Maneuvering with sidewinder delivery or air-to-air gunnery. Counter tactics against surface to air missiles and anti-aircraft artillery can be provided as a part of the WTT training exercise.

Both OFT and WTT trainers have the capability to provide instant replay, or simulation freeze for the student as an instructional tool to emphasize a particular situation during the training session. In addition, full video debrief capability is required for off-line debrief and analysis of the training session.

DESIGN CONCEPTS

Prior to this point this paper discussed the training needs of the AV-8B community. These training requirements have been thoroughly evaluated to determine the performance requirements of the resulting trainers. These performance requirements have driven a design concept that generates a suite of advanced technology simulators which will be available for training of pilots in early 1985. The following discussion presents some of the unique characteristics and capabilities of this simulation system.

VISUAL SYSTEM

For both the OFT and the WTT the visual system is a significant advancement in the capability of fighter/attack training devices. At the heart of the visual system is an advanced computer image generation system. Figure 1 shows typical computer image generation scenes. This system is to provide the visual environment necessary to meet the training requirements. This environment consists of a digital data base stored on a 200 M byte disc. In this case the data base covers a 400 nautical mile by 400 nautical mile area surrounding Cherry Point, North Carolina from sea level to 45,000 feet. Since the training requirements of the OFT and WTT are different the data base model, although essentially the same geographic area, is tailored to the specific needs of each individual simulator. For example, the OFT is intended for take-off and landing with navigational capabilities. Therefore, system resources are concentrated near airports and along specified navigation corridors with scene content modeled to support flight in these regimes. The WTT is intended as a weapons delivery trainer and, therefore, the data base is optimized with low altitude navigation and target penetration with specific weapons delivery regions that are scene rich in detail and include an assortment of fixed targets as well as moving targets.

The OFT data base will provide specific geographic modeling in areas where those cues are related to a required training task. Other areas of the data base are "filled in"

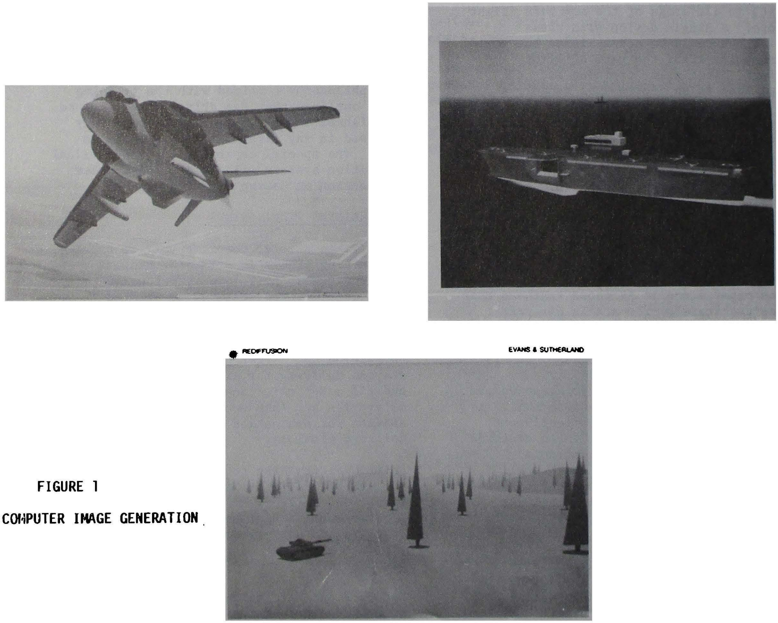


FIGURE 1

TYPICAL COMPUTER IMAGE GENERATION

with generic forest and fields as appropriate. This technique allows much higher levels of detail in these areas without expending disproportionate levels of system resources. The OFT data base will also contain ships in the ocean off the coast. These ships, modeled as dynamic coordinate systems under host computer control include wake turbulance and sea state.

Special features of the image generation equipment include Visual Approach Slope Indicator (VASI), Fresnel Lens Optical Landing System (FLOLS), AV-8B shipboard visual landing aid and other types of lighting such as rotating beacons and approach strobe lights. Weather-related special effects are also a capability of the visual system. These effects include cloud and fog simulation with ambient visibility computations to simulate haze conditions. Special effects such as landing light glare and anti-collision lights as a function of fog/cloud density are included. All of these effects are properly controlled as a function of illumination levels with sun angle computation for shadowing effects. Six different levels of natural illumination provide time of day simulation.

The WTT image generation equipment has the same capability as the OFT but its a data base that has been optimized for weapons delivery regions. These target areas include SAM/AAA sites some of which are mobile and some fixed. Up to three moving targets can be displayed simultaneously, including fixed wing aircraft, helicoptors, tanks, trucks, and trains.

DISPLAYS

The weapons delivery tasks demand a large Field-of-View to adequately train pilots in the simulator. In order to provide a large field-of-view without objectionable gaps or overlaps between channel boundaries a real image projection scheme was developed. For the WTT it consisted of seven full color raster projectors with wide angle optics in a continuous mosaic display. Figure 2 shows the field-of-view plot for the WTT display system. The projector selected was a General Electric high intensity, high resolution color light valve projector. In order to accommodate the computer image generation equipment overload management technique of field rate extension, the light valve projector had a design modification installed to maintain high quality imagery during overload conditions. The mosaic display offers a wide field-of-view system but requires that each channel be carefully pieced together to provide a continuous display without any discontinuity at the channel boundaries. Therefore, a series of precision mechanical masks with fine tuning position adjustments were developed. These masks located at an image plane within the custom designed wide angle optics provide the capability to precisely locate the channel boundary on the display screen. This scheme provides a continuous visual scene without channel overlaps or gaps.

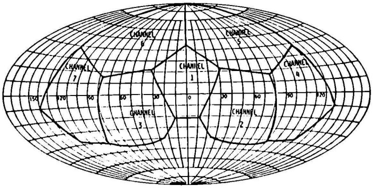


FIGURE 2

WTT FIELD-OF-VIEW DISPLAY PLOT

The next consideration for the display system was that of geometric distortion. Whenever a real image projection system is employed, the viewer must be "on-axis" with the optics to prevent distortion. Locating all seven projectors in the case of the WTT, on-axis with the cockpit eyepoint is not feasible therefore, the projectors must be located "off-axis" and a method of introducing distortion of an equal but opposite magnitude must be implemented. This correction pre-distortion takes the form of a linear "keystone" shaped function. This pre-distortion can quite easily be accommodated in the computer image generation equipment simply by mathematically defining the channel window as tilted with respect to the view axis rather than orthogonal as in a conventional application.

The second part of the distortion problem was that of projector/optics distortion. Most optical designs will deviate from a linear tan mapping as the optical half angle is increased. Since the distortion introduced in the optics results in image discontinuity at the channel boundary this distortion must be corrected as well. In the design of the wide angle optics the transmission, resolution and distortion parameters were all carefully evaluated to provide optimum performance. Since the distortion could be corrected, the brightness and resolution parameters were optimized at the expense of linearity.

The optical distortion was then corrected by a raster shaping technique within the light valve projector. Due to the requirement for precision channel matchup a series of correction equations for raster-shaping were developed and implemented into the hardware. Due to the number of projection systems and the time required to align these projectors by hand, an automatic computer controlled raster alignment scheme was developed. The host computer calculates the correction equation coefficients which minimizes the displayed error in reference to calibration lights embedded in the display surface. These coefficients are then digitally downloaded into the respective projector to correct distortion errors. This allows quick reference to projector performance, geometric accuracy and fast alignment if necessary.

The General Electric light valve uses a single gun system to provide inherently color converged displays. This single gun system utilizes a set of dichronic filters which break down the white light from the Xenon arc lamp into green and magenta light. The green light is modulated in the vertical axis of the raster and the magenta light is modulated in the horizontal axis to produce blue or red light as a function of the frequency of modulation. In this manner complete control of an R,G,B system is obtained without the convergence problems of a three gun system. Since the operation of this system is based on deflection modulation for the green and magenta colors being orthogonal, introducing a non-linear raster shape causes severe color distortion. These color distortions can be corrected within the projector and are the third set of modifications included in the OFT/WTT light valve projectors. The result is a high quality, extremely accurate wide field-of-view display.

The OFT then incorporates the same set of projectors and image generator with five channels instead of seven since the required field-of-view is somewhat smaller than the WTT. Figure 3 shows the field-of-view plot for the OFT display system.

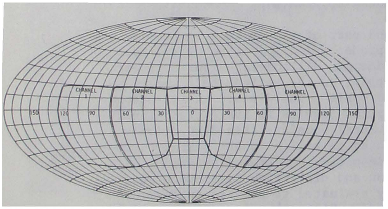


FIGURE 3

OFT FIELD-OF-VIEW DISPLAY PLOT

TARGET PROJECTORS

The WTT has a pair of very high resolution, slewable target projectors. This additional capability is provided for both air-to-air combat and air-to-ground weapons delivery where high resolution imagery is required but can be accomplished by a limited field-of-view projection scheme. For example, an air target requires high resolution but generally subtends a relatively small angle at the design eye. A pair of these projectors are utilized in order to provide coverage for the entire display area without being located within the pilots field-of-view. These projectors are primarily intended for displaying air targets, either fixed wing or helicopter, but can also be used to display ground targets for long range acquisition and identification.

CREW STATION

The AV-8B simulator crew station is an exact replica of the actual aircraft. Extensive use of flight hardware preserves the illusion of realism for the student. Figure 4 shows a photograph of the trainer crewstation. The ejection seat has been modified into a pneumatic G-cueing system. The seat pan is equipped with four independently computer controlled cushions. These cushions can be used to generate sustained positive or negative G load cues. Positive loads are simulated by the student "sinking" into the seat while negative loads cause the student to raise. Further negative load cues are provided by a differentially driven lap belt to generate the sensation of hanging by the belt. In addition the seat backrest is configured with three pneumatic cells which provided positive and negative longitudinal acceleration cues. These cushions can all be independently computer controlled for roll/sideslip cues. The buffet cushion provides multiple complex frequencies up to 20Hz to accurately reproduce the aircraft buffet sensation.

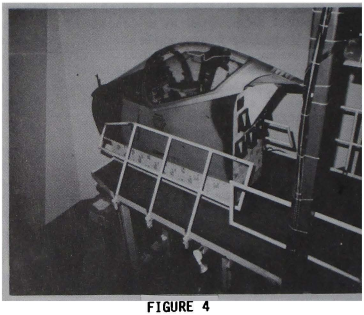


FIGURE 4

AV-8B TRAINER CREW STATION

All of the G-cueing systems described above are time correlated with the visual system to re-enforce the sensation of motion. The sustained G load simulation described above is followed by a continuous smooth reduction of all display intensities and lights to simulate pilot blackout.

To further re-enforce the visual and motion cues a high fidelity sound simulation system accurately reproduces the aural environment of the AV-8B aircraft. The environment consists of gas turbine starter, engine ignition, engine sounds including failures such as compressor stalls, landing gear transients, flap noises, windstream noises and many others. In addition to the environment the full aircraft complement of tones are provided.

The radio simulation system fully reproduces all navigation/communication capabilities of the AV-8B aircraft radios. The student can be trained in radio navigation throughout the data base including TACAN station identification tones. Contact can be established with any of the surface facilities while the instructor monitors student manipulation of aircraft radio systems. The instructor has the option of establishing instant communication with the student if the student is not properly utilizing the radio system. The Automatic Terminal Information Service (ATIS) and Ground Controlled Approach (GCA) is a completely automatic computer controlled voice digitizing system that frees the instructor from the task of playing the role of ground based controller. This system employs a high density winchester disc to store a specified vocabulary which the host computer can pull out of memory to assemble a high quality message which is then converted to audio and transmitted to the student through the radio simulation system. The KY-58 secure voice encoder is also a part of the radio simulation system and provides students the opportunity to become familar with its operating characteristics.

INSTRUCTOR/OPERATOR STATION

The instructor station allows complete student monitoring through the use of color visual system repeaters. The instructor station is designed to accommodate either a single instructor or an instructor and operator together. Figure 5 shows the OFT instructor station in assembly. Cockpit heads-up display imagery is also repeated and optically superimposed over the center channel repeater display with the same accuracy as the trainee station displays. The cockpit Digital Display Indicator (DDI) is also available at the instructor station. All of these displays plus audio are recorded on a video cassette player for a full video debrief capability off line.

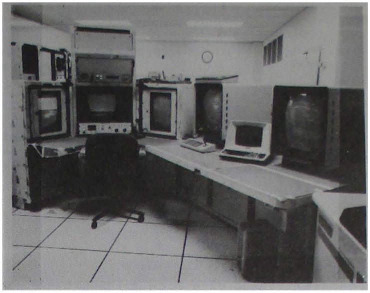


FIGURE 5

OFT INSTRUCTOR STATION
(IN ASSEMBLY)

The graphics system displays provide instructor interaction capability with the host computer for set-up and monitoring of each training task. All cockpit switches, levers, instruments, displays and lights are available at the instructor station for monitoring. To minimize the complexity of the instructor/computer interface the graphics displays have infrared touch screens built in. These displays incorporate software prompts to lead the instructor through the various CRT pages required for problem set-up and monitoring. Through the graphics displays, the instructor can insert faults and monitor student reaction to emergency conditions. The instructor can control environmental conditions such as weather/visibility, time of day, sea state, wind velocity, turbulence, etc.

CONCLUSION

The training systems described in this paper are intended to fulfill a specific need for the Marine Corp of the mid 1980's and beyond. Their requirement is to provide high fidelity training to pilots transitioning into the AV-8B Vertical Take-off and Landing aircraft. The trainers described here advance the state of the art of fighter/attack simulators and increase the pilot proficiency in all areas of training while minimizing the risk of pilot and equipment loss during the early phases.

85-1738

SIMULATING THE WORKS: FROM CREW INPUT TO SYSTEM RESPONSE

Richard A. Weeks*
Northrop Corporation
Advanced Systems Division
Pico Rivera, California

Abstract

In the development of a highly integrated airborne weapon system, many types of research and development simulations are used. These simulations range from simple off-line analysis types to extremely complex, real time, crew-in-the-loop simulations. In the realm of those which are real time, a variety of approaches also exist. These approaches may vary from simple system definition or design verification experiments, to part task or mission segment, simulations to full mission simulation.

In full mission simulation, emphasis is placed on the modeling of all system functions, accounting for critical crew/environment stimuli which reflect changes in the system state. In a sense, it is simulating the works, the entire airborne weapon system (including the crew). In the creation of the simulation, a careful examination of crew interactions with various aircraft subsystems and of the expected system responses must be made. This examination is accomplished relative to the test requirements so as to properly determine the degree of simulation fidelity needed to support the development of the system design.

This paper will present design concepts and examples relative to the creation of full mission simulations for highly-complex, integrated, airborne weapon systems. Airframe and flight control modeling will be addressed with discussions dealing with the coupling of avionics systems. Concepts relating to the various avionics system simulations will highlight this paper with analyses of controls and displays, flight-oriented and mission-oriented avionics. Vehicle subsystem modeling will also be presented to address fidelity issues regarding the coupling effects between avionics and vehicle subsystems. Systems monitoring and multiplex bus simulation concerns will also be presented in this paper.

Weapon System Dynamics

Full mission simulation places an emphasis on real world scenario replication. Just as a control-system filter response is based on the dynamical qualities of the process and the input, the Airborne Weapon System is concerned with the input and system dynamics.

The input, in the case of the weapon system, is the crew and the external environment. The environment not only affects the system response directly but will also influence the crew to perform tasks in a certain manner. The system dynamics, on the other hand, deal with the behavior of the system functions and operations. These system functions and operations, when stimulated by crew/environment inputs, induce a system response indicative of the dynamical properties of the system. It is these system responses which are investigated in the simulator. This investigation is made possible by faithful representation of the system (or system design concept) functions and operations. As the crew operates the various systems in the simulator, a proper stimulus is generated since the proper environment is simulated.

System Development Using Simulation

During the development cycle of a sophisticated airborne weapon system, many levels or types of simulation are used. These types range from very elementary system definition types to the elaborate full mission simulation. During the utilization of these simulations, the system under development matures. This process is shown in Figure 1.

System concepts for various weapon system implementations or designs originate within cognizant engineering design agencies where critical requirements and objectives are generated. Following the generation of these preliminary requirements, traditionally a number of paper design studies and/or research/analysis is accomplished with the resultant generation of one or more candidate approaches or solutions to meeting the system requirements. To support the evaluation of these candidate design approaches/concepts, a number of subsystem definition simulations are generated. These simulations may be either real time or non-real time and offer little (if any) realistic man-machine interfaces. That is, evaluations with these type of simulations might not be performed using a simulator cockpit, but might simply use off-line display devices in a computer laboratory. In fact, this level of simulation is more concerned with the functions performed by the subsystem being designed rather than the crew interaction with the system.

Following the completion of these subsystem definition simulations, more comprehensive critical requirements and functional definition of the subsystem can be generated. At this time, real

*Member, AIAA and AIAA Flight Simulation Technical Committee

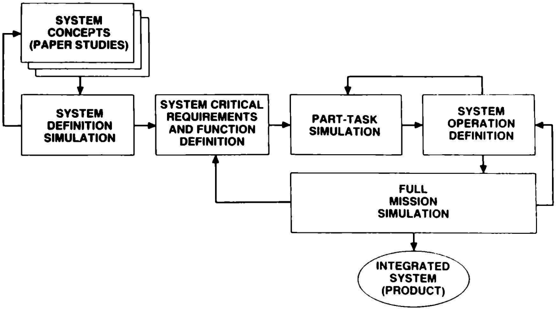

SYSTEM CONCEPTS (PAPER STUDIES)
SYSTEM DEFINITION SIMULATION
SYSTEM CRITICAL REQUIREMENTS AND FUNCTION DEFINITION
PART-TASK SIMULATION
SYSTEM OPERATION DEFINITION
FULL MISSION SIMULATION
INTEGRATED SYSTEM (PRODUCT)

Figure 1. Avionics Development Using the Simulator

time part-task simulations of the more favored subsystem design concepts are implemented for crew evaluation. Additionally, the system operational characteristics are examined. Here, a realistic crew environment is required since the acceptability of the candidate design is being evaluated for optimality of the crew interaction and operation. As is often the case, there frequently is a redesign/retesting of the system operation using the part-task simulations. This redesign is required due to the candidate system′s failure in meeting the design requirements, including ineffective crew control/monitoring of the subsystem.

Subsequent to the refinement of the various subsystem operations and performance characteristics using the part-task simulations, an integrated full mission simulation is constructed. This is accomplished by integrating the basic part-task simulations together and by adding appropriate subsystem-to-subsystem interfaces. The use of full mission simulation then allows the evaluation of the total system concept, addressing operational effectiveness relative to the mission objectives. Here, it is extremely important to provide realistic mission environments and crew interfaces, to assure the system/subsystems are exercised in the manner intended in actual usage. During testing activities with full mission simulation, it often becomes necessary to redefine some subsystem operations and/or the subsystem critical requirements. Upon completion of the full mission simulation activity, an effective, integrated system would evolve.

Key Concepts For Mission Simulation

In the development of full mission mission simulations, several key concepts must be realized. They are:

- Operational fidelity
- Proper task loading
- Correlation

When constructing the simulations, operational fidelity is maintained by judicious application of effective mathematical modeling techniques when representing the characteristics of the weapon system. Operational fidelity relates to the accurate replication of crew interfaces with the various vehicle system simulations. There must be no perceptible differences between the operational characteristics of the weapon system represented in the simulator, relative to the actual aircraft. In the case of new systems development, the system simulations must operate like the system design concept. Pilot evaluations and flight test data (if available) are typically used to verify and validate the operational fidelity of the simulator.

By providing the realistic or operationally faithful representations of the various subsystems, as discussed above, proper task loading is achieved. To attain proper or a realistic task loading, the crew must work in the simulator in a manner identical to that in the actual aircraft. This is particularly critical when the simulator is being used to develop crew interfaces on a new system. Proper task loading will allow the crew evaluating the system to accurately assess the operability and effectiveness of the system.

Lastly, but most importantly, everything in the weapon system simulation must correlate. That is, the entire simulation has to produce an effect which is consistent with the real world. This pertains to all environmental stimulus to the crew and the interactions of the various subsystems with regard to all aspects of the mission being simulated. Correlation is only achieved by designing the simulation with that intent. It is not an easy goal to attain and generally requires significant effort to produce a real world correlated simulation, particularly if imaging sensors and the outside visual reference is represented in the simulator. However, in constructing an operationally faithful simulation with accurate task loadings on the crew, the first step in correlation is produced.

Modeling of the Airframe

Much of the effort on airframe modeling is accomplished primarily in support of flight controls development or handling qualities evaluations. Many times, the simulations representing the airframe use hundreds-of-thousands of values of aerodynamic data parameters. This is necessary when full envelope maneuvers are being evaluated and when flexible structural modes (many degrees-of-freedom) are represented.

In the application to full mission simulation, airframe response modeling is quite important. Since a full mission simulation encompasses scenarios from take-off to enroute navigation to low-level terrain following to aerial refueling, full envelope airframe mechanizations are required. Additionally, the acceptability of vehicle subsystem operations or avionics system responses can be traced back to the relationships of the subsystem to the airframe model. For example, in development of air-to-ground weapon delivery algorithms, aircraft performance/state data is typically used with the algorithms to determine weapon impact or release points. Proper airframe mathematical representations will therefore enable proper system and subsystem interfaces.

At Northrop, Advanced Systems Division, full mission simulation programs typically "baseline" various configurations of the airframe models which are generated initially for flight controls development. This baselining is accomplished at various stages of aircraft development to insure that the latest validated (and pilot-approved) airframe representation is used during mission evaluation.

Vehicle Subsystem Simulation

The simulation of the various vehicle subsystems typically range from those which are simplistic to those which are moderately complex. The degree of fidelity, in general, depends on the functions to be evaluated or the crew interfaces to be optimized.

As an example, propulsion subsystems are generally modeled in high detail to support flight controls development. Engine dynamics and thrust response are key issues in the modeling process since the pilot has a direct control input to the system and will react according to the resulting engine response characteristics.

Fuel subsystems are typically modeled with varying degrees of realism. Simple center-of-gravity (c.g.) effects of fuel loading can be treated simplistically as a discrete variable in the simulation, or can be modeled by a faithful replication of the fuel system. The latter approach accounts for fuel transfer/flow characteristics of the subsystem which provides a continually changing c.g. effect or allows for the evaluation of the suitability of autonomous fuel/c.g. management systems.

The modeling of various other vehicle subsystems such as electrical, hydraulic, air data, environmental control, landing gear, etc. are also accomplished with varying degrees of fidelity. Typically, enough detail is modeled such that realistic operator interactions with the subsystem simulations (through the crew station controls and displays) is attained. Additionally, the effects of malfunctions or failures of various elements within the subsystems are provided when degraded operational system modes are to be evaluated.

When system failures are to be modeled in the simulation, precautions must be taken to insure that the proper intra-system, as well as inter-system communications and responses are considered. For example, an engine malfunction affects many other subsystems in a dramatic way, such as partial loss of hydraulics and electrical power. More subtle failures, however, such as the loss of a single hydraulic pump might simply degrade the landing gear subsystem braking forces or flight control actuator maximum deflection rate or authority. These latter degraded effects might go undetected by the crew or may only slightly alter the vehicle performance characteristics. Nonetheless, in the generation of a full mission simulation, all operations of the vehicle subsystems must be properly modeled, with sufficient fidelity, so as to evaluate the overall system design.

Avionics System Modeling

The simulation of avionics systems becomes extremely critical in the generation of a full mission simulation. With the continual development of sophisticated, high-speed, airborne digital processing equipments, avionics systems continue to become more complex and intricate. New system architectures are being implemented which allow each avionic subsystem to process localized functions and control other subordinate elements of the system while reacting to the commands from a centralized controller. In simulating these architectures, the representation of digital multiplex bus traffic becomes important when determining the acceptability of the system reaction time (latency) under particular loading conditions.

When developing effective integrated avionics systems, it is desired to cast the crew into the role of systems manager such that the operation of the system will require minimal effort. In this manner, mission goals are more easily obtained. The use of full mission simulation supports the allocation of the crew functions, the assessment of the workload and the optimization of the design such that this systems manager role is attained. In the development of the supporting avionics simulation, maximum fidelity is required.

The simulation of various types of avionic subsystems and processes is accomplished when full mission scenarios are developed for crew evaluation. In general, core avionics (such as MIL-STD-1750, MIL-STD-1589, etc.) are not simulated. Since these cores deal with airborne operating system, multiplex bus protocol, and higher-order language standards, the simulation system equivalents (such as FORTRAN) are generally used. This is accomplished without compromising the integrity of the evaluations of the operational characteristics of the system being developed in the simulator. Core avionics simulation (or emulation) might be used during hardware and operational flight program development rather than during system concept development which is accomplished in the simulator.

The simulation of flight-oriented avionics is always included in the development of a full mission simulation since these types of systems are a key element in determining the system transfer function. Flight-oriented avionics include flight management systems which couple responses between the crew and the flight control system, navigation systems, interfaces to the various vehicle subsystems (especially fault monitoring) and crew station Controls and Displays.

The simulation of Controls and Displays is extremely critical since the crew environment must be properly represented. Accurate replication of the switchology implementations for the various aircraft subsystems must be maintained. The system output functions, in the form of cockpit displays, are of paramount importance and must be realistically portrayed. For crew stations employing the use of electronic displays (which is quite typical), the symbolic display formats and display generation must faithfully represent the actual system or the system design concept. This requirement may be very difficult to accomplish (without the use of real aircraft hardware), especially if displays are simulated using general purpose graphics generators. Electronic display simulation is even more complex if a hybrid-type display format is required. Here, both stroke (or calligraphic) and raster graphics are combined on a single Cathode Ray Tube, which requires intricate timing and synchronization. Proper simulation of brightness, resolution and display writing characteristics must be addressed.

Lastly, the simulation of mission-oriented avionics is critical to the development of the weapon system. Mission-oriented avionics includes systems such as radar, stores management, defensive management, etc. These types of systems are usually quite complex and require an extensive amount of crew evaluation in the simulator in order to produce an effective product. Consequently, this type of avionics is modeled in extreme operational detail to support these critical evaluations.

When developing the simulations of a particular system (such as a radar system), it is important to realize that all modes of operation of that system must be accounted for. The transition logic and interaction of the various modes is just as critical as the operating dynamics of the mode itself. This is particularly true when crew involvement is required in the mode selection process.

Mission Environments and Simulator Systems

Simulating the mission environment is a key element when evaluating the effectiveness of an airborne weapon system. During all phases of flight, the appropriate outside visual reference must be presented without compromise. For most portions of avionics evaluations, a simple sky/earth horizon reference is adequate. However, for handling qualities evaluations (during takeoff/landing or aerial refueling, for example) or during visual air-to-surface weapon delivery, a more detailed external visual scene is required.

At Northrop, Advanced Systems Division, an Evans and Sutherland CT5A image generator is used which produces high resolution video images of the external visual environment of the real world. Figure 2 shows an illustrative image produced by a CT5A image generator. The on-line CT5A data base is representative of the Northeast United States where Griffiss and Plattsburg Air Force Bases are accurately modeled. Many real world features around the airbases are represented in the data base, including cities, main roads, rivers/lakes, railroads, etc.. At the airbases, all runways/taxiways, hangars and buildings are modeled. It is possible to land on the runway, taxi to a hangar and park the simulated aircraft. All lights at the airfield (VASIs, strobes, landing lights, beacons, etc.) are also modeled with illumination intensity controlled by the simulated time-of-day or as commanded by the simulation computational system. Several terrain corridors are also modeled which permit long duration terrain following flight. In all portions of the visual data base, three-dimensional texturing is applied, offering critical height and speed cues to the pilot. This cueing is particularly important in the airbase areas during takeoff and landing and in the terrain corridors for low-level flight.

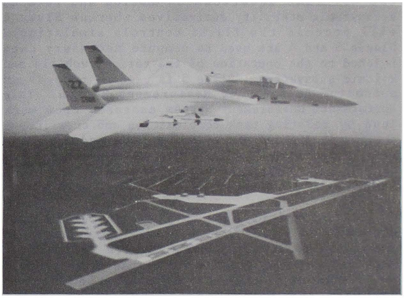


Figure 2. Real Time CT5A Image

In support of the development of advanced, high-fidelity simulations, a modern computational system is required. This system must be capable of processing the simulation software which represents the various aircraft subsystems, avionics, flight controls and aircraft dynamics in an effective, real time fashion. At Northrop, Advanced Systems Division, a Harris computational system is used.

This Harris system is comprised of Harris 800 computers which share a common memory. This configuration is shown in Figure 3. Although 6 processors are used, only one of them (designated the Master) retains a full operating system. The other 5 act as peripheral "slave" processors to the Master. This arrangement provides for an optimum real time utilization.

Each of the 6 Harris computers process a part of the overall mission simulation software. For

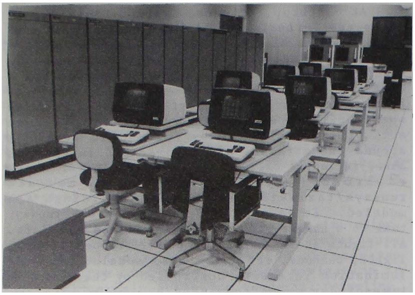


Figure 3. Harris Computational System

example, the Master will provide communications with peripheral processors used in the simulation, such as the graphics processors for in-cockpit displays and switchology decoding microprocessors for the crew station. Slave 1, on the other hand, will execute the functional software representing the airframe equations-of-motion and related aerodynamic stability derivatives whereas Slave 2 will process the flight controls simulations. Slaves 3 and 4 are used to compute necessary data related to the operation of the various vehicle and avionic subsystems. Slave 5 is used to process an automated Performance Measurement System. This Performance Measurement System assesses crew work load by relating task complexities and crew interactions in the cockpit with controls and displays equipment designs. This System is designed and implemented by Northrop, Advanced Systems Division Avionics Engineering.

Peripheral to this "6-plex" of Harris computers are a variety of graphics generators and special purpose processors. Twelve Interactive Machines Incorporated color stroke graphics systems, 4 Megatek color, high-resolution raster graphics systems, 1 Adage 3000 color, high-resolution raster graphics system, 1 Adage 4135 monochromatic stroke graphics system and 4 dual-channel stroke/raster (hybrid) processors are used to drive various displays in a simulated crew station and at an operator´s console. Additionally, a Floating Point Systems array processor and Electronic Associates Incorporated hybrid linkage are also tied to the Harris computers to perform specialized functions.

An operator´s console is used to control and monitor the simulations which are being executed on the Harris computational system. A variety of color displays are used for this purpose. Some of these monitors have clear touch-sensitive panels overlaid on them such that graphical control menus can be displayed. From these menus, an operator merely touches the screen at the particular menu item position describing a control function, and that control function becomes invoked. This method of control offers flexibility in the control of a complex simulation and is capable of handling thousands of input/output control functions.

Another critical feature of a full mission simulator is the flight simulator itself. The flight simulator provides the cockpit environment for the crew to operate during system evaluation. This cockpit must be representative of the actual aircraft so as to satisfy the proper task loading/operational fidelity requirements described earlier. The flight simulator may also provide the motion/aural cues required during various phases of flight. Northrop has used a variety of moving-based and fixed-base flight simulators in the development and evaluation of various weapon system designs.

Northrop, Advanced Systems Division´s latest simulator development uses a six-degree-of-freedom synergistic motion base on which a high-fidelity crew station is mounted. This flight simulator also uses a Rediffusion Wide Angle Infinity Display Equipment (WIDE) system for presentation of the outside visual scene as generated by the CT5A image generation system. The support structure for WIDE and the motion base platform allows for the interchanging of complete crew stations. This interchangeable crew station concept is based on the use of microprocessors in the crew station itself. All switchology inputs and non-electronic display outputs are processed by this microprocessor system. These microprocessors communicate with the Harris via a high-speed parallel link. Therefore, when a cockpit is to be interchanged, wiring disconnects are minimized since most cockpit input/output is established through this digital link.

Lastly, another key simulation system used in the support of full mission simulation is that which is used for imaging sensors, such as ground mapping radar, electro-optical or infrared. The digital simulation of these systems is extremely difficult and may be very difficult to correlate with the outside visual scene, depending on procedures employed during development of the supporting digital data bases.

At Northrop, ground mapping radar simulation has been accomplished in the past with a photomosaic model board, and more recently in a simplistic fashion by the use of actual radar imagery stored on digital disc. This approach offers extreme realism but is inflexible and maintains little or no correlation with the outside visual scene. This method is supportive of head-down target recognition experiments.

To provide a more realistic radar capability, Northrop´s Advanced Systems Division is in the process of procuring a Digital Imaging Radar Simulator. This system will provide extremely detailed radar video images with a one-to-one correlation capability with the CT5A image generation system. Furthermore, this system will be fully digital and will offer maximum flexibility for modifying radar parameters for conceptual system design analysis.

Summary

In summary, the generation of a full mission simulation capability provides an extremely useful tool in the development of a complex, airborne weapon system. This approach allows extensive crew evaluations of the system being developed, early in a development program, which can enhance the optimization of the total system. The system is characterized by responses to crew/environment inputs and the functional operations of the system.

In the development of the simulation, a varying degree of modeling fidelity is encountered. Some models, such as the Flight Control System and Avionics are provided in high detail due to the safety-of-flight and mission success issues, respectively. Other models, such as the vehicle subsystems, might not require such a high level-of-fidelity. To support the generation of the mission environment, careful consideration must be given to the simulator systems required to satisfy the requirements of maintaining a faithful representation of the environment.

85-1740

OPTIMAL GENERALIZED MULTISTEP INTEGRATION FORMULAE FOR REAL-TIME DIGITAL SIMULATION

Daniel D. Moerder and Nesim Halyo
Information & Control Systems, Incorporated
28 Research Drive
Hampton, VA 23666

Abstract

The problem of discretizing a dynamical system for real-time digital simulation is considered. Treating the system and its simulation as stochastic processes leads to a statistical characterization of simulator fidelity. A plant discretization procedure based on an efficient matrix generalization of explicit linear multistep discrete integration formulae is introduced, which minimizes a weighted sum of the mean-squared steady-state and transient error between the system and simulator outputs.

1. Introduction

This paper considers the problem of simulating the trajectory of a dynamical system described by

$$\dot{x}(t) = Ax(t) + Bu(t) \quad x(t)\epsilon R^{n} \; u(t)\epsilon R^{m} \tag{1}$$

$$y(t) = Cx(t) \quad y(t)\epsilon R^{\ell} \tag{2}$$

on a digital computer in real time, where y(t) is the system output and u(t) is some external input signal. We define a real-time simulation as having the property that passage of time in the simulation takes place at the rate as it does in the environment.

A general Real-Time Digital Simulation (RTDS) operating in, and responding to inputs from its physical environment introduces several special concerns:

1. Define an integration stepsize as T > 0. In real time, the propagation of the state x(t) to x(t+T) must be completed at time t+T. While general "closed" or implicit integration formulae take the form

   $$x(t+T) = f(x(t), \dot{x}(t), t, T; \dot{x}(t+T))$$

   the real-time formula must satisfy

   $$x(t+T) = g(x(t), \dot{x}(t), t, T)$$

   It is well-known that these latter "open" formulae have accuracy and stability characteristics which are dramatically inferior to those of comparable closed formulae of equal order and stepsize. In particular, note that formulae suitable for real-time implementation require at least a full integration step before they can respond to system inputs.
2. The RTDS is a sampled-data system with a sampling interval T which is fixed by the computational speed of the implementation hardware. In order to accommodate input signals with unrestricted frequency content, it is necessary to place an antialiasing lowpass filter in convolution with the input in order to bandlimit it. The prefilter dynamics introduce a priori phase lag and distortion into the simulation. In addition, due to engineering tradeoffs and noise, the bandlimiting is never really perfect, so that there is usually some aliasing distortion of the input.
3. Output reconstruction (D/A conversion) must be done using a causal data hold. The continuous-time approximation of the system output between t and t+T must be a function only of data at t or earlier. This introduces still further phase lag and distortion.

The above concerns can be neutralized by allowing T→0 and discarding the input prefilter. This "brute force" approach, however, is impractical for simulations of all but trivial simplicity. Implementing the large, elaborate computer software in modern piloted simulations, for example, generally limits the sampling frequency to a range between 10 and 30 Hz. At these low sampling frequencies, the distortions discussed above significantly degrade the stability and performance of the RTDS.

It is widely recongnized that classical general-purpose causal numerical integration schemes are ill-suited for use in digital simulations, primarily because of inefficiency and/or the introduction of spurious parasitic modes [1,2]. The past twenty years has seen a number of studies [3-8] aimed at developing improved procedures for deriving discrete models of continous-time plants. The procedures developed in [3-5] are primarily aimed at controlling the frequency response of the discretization. In [3], a procedure is derived for synthesizing a discretization which minimizes a quadratic measure of the disparity between the poles of the discretization and those of the z-transform of the continuous plant transfer function, given that the discretization includes some dynamics which cannot be varied. In [4], (the well known "IBM Method") an adaptation of root locus techniques from classical control theory is applied to the design of simulations consisting of a plant discretization with output reconstruction. This method has had successful practical application, but requires extensive analytical effort. Reference [5] describes a class of tuneable digital integrators in which phase and gain characteristics of the discretization can be empirically adjusted to achieve desired response.

The discretization design problem has also been examined from other points of view. A procedure is given in [6] for optimizing the stability region, or maximum stable integration stepsize, of a linear multistep integration algorithm. References [5] and [7] examine the digital simulation of nonlinear systems with multistep algorithms using transition matrix-like expressions based on local linearization of the dynamics. These latter procedures have had considerable success in simulating complicated rotational motion in real-time. Reference [8] describes an approach to deriving linear multistep integration procedures in which, rather that being derived based on a polynomial approximation to the state derivative time history - as in the case of classical formulae - the formula is based on a sinusoidal approximation. The method is aimed at simulating systems whose response in dominated by oscillatory dynamics.

In Section 2, a statistical approach to characterizing simulation fidelity is described, which reflects the entire ensemble of dynamic elements comprising the RTDS, and an analytically tractable scalar figure of merit is defined. This fidelity measure is applicable to multivariable simulations, and reflects the full range of transient and steady-state dynamic phenomena in a generic RTDS.

The measure appears in Section 3 as an optimization criterion in a systematic procedure for designing plant discretizations. This procedure is of considerable practical interest. Aside from the fact that it rigorously accounts for the full end-to-end simulator dynamics, it is easily mechanized. This feature is of considerable importance when the plant dynamics are subject to frequent change.

Section 4 provides a numerical demonstration of the design method and Section 5 presents conclusions.

2. RTDS FIDELITY

This Section describes the configuration of a generic RTDS for the system (1,2) and identifies sources of random and structural output signal distortion. This discussion leads directly to the derivation of an analytically tractable expression which can be used to characterize RTDS fidelity.

Figure 1 schematically displays an idealized RTDS for the plant model (1,2). The input signal u(t) is sampled with sampling interval T before passing into the simulation. It is assumed that u(t) can be modelled as

$$u(t)=C_u x_u(t) \tag{3}$$

$$dx_u(t)=A_u x_u(t)+B_u dw(t) \quad x_u \epsilon R^{n_u} \quad w \epsilon R^{m_u} \tag{4}$$

where the dynamics in (4) are asymptotically stable and where w(t) is a weakly stationary zero-mean Gaussian process with

$$E\{w(t)w^T(s)\}=W\,\delta(t-s) \tag{5}$$

Next, the plant trajectory is approximated using a discretization of the system (1). Finally the simulator output $\hat{y}(t)$ is recovered in continuous time through some causal reconstruction strategy.

By Shannon's sampling theorem, this ideal case includes an implicit assumption that the frequency spectrum of the input signal is limited to a bandwidth of 2π/T, henceforth referred to as the fundamental band. In general, the frequency content of y(t) from (2) contains components both inside and outside the fundamental band, so that the system output can be decomposed as

$$y(t)=y_s(t)+y_h(t) \tag{6}$$

where $y_s(t)$ is that portion of y(t) whose frequency content is contained in the fundamental band, and $y_h(t)$ is the remainder. It is assumed that it is of interest to simulate only $y_s(t)$. Because of this, and because $y_h(t)$ disappears for sufficiently small T, $y_h(t)$ is ignored in the sequel. The sources of error in the idealized RTDS are nonrandom, stemming from the difference between the dynamics of the original system (1,2) and that of the convolution of the discretization and associated reconstruction strategy in the RTDS.

A more realistic model for an RTDS is shown in Figure 2. One conspicuous nonideal feature shown here is the presence of an input prefilter. Since the input sampling interval is generally determined by the computational speed of the digital computer used to implement the discretization, there is no guarantee that frequency spectrum of the input signal will be matched to the sampling interval. Thus, it is necessary to pass u(t) through an analog filter which excludes input frequencies outside the fundamental band.

Even before its passage through the prefilter, the input signal is corrupted by noise sources associated with data communication lines, the power supply and so forth, represented by n(t) in the Figure. For this discussion, these effects will be modelled as a weakly stationary zero-mean Gaussian process with covariance

$$E\{n(t)n^T(s)\}=N\delta(t-s) \tag{7}$$

The transfer matrix of the prefilter has the realization

$$u_f(t)=C_f x_f(t) \qquad u_f \epsilon R^m \tag{8}$$

$$dx_f(t)=[A_f x_f(t)+B_f u(t)]dt+dn(t) \quad x_f \epsilon R^{n_f} \quad n \epsilon R^{n_f} \tag{9}$$

Naturally, it is assumed that the prefilter is asymptotically stable. As with the case of the input signal dynamics, we are considering only strictly proper prefilters. This is only done for simplicity, since the extension of this theory to the case of proper prefilters (such as the type-2 Chebyshev) is straightforward.

Further corruption of the input signal takes place after prefiltering. Due to engineering tradeoffs between bandlimiting and signal distortion, the prefiltered signal will not be perfectly bandlimited, so that sampling this signal will induce some aliasing. It has been shown [9] that when u(t) and n(t) are both zero-mean Gaussian random processes, the aliasing is a zero-mean Gaussian process, statistically independent from the fundamental signal. This aliasing noise is represented as $u_a(t)$ in the Figure.

In analog/digital (A/D) conversion at the interface with the digital computer, the sampled signal is truncated or rounded into a digital number sequence. This introduces quantization error, $n_q(kT)$ in the Figure. It is assumed that the quantization error is small relative to the typical signal variation from sample to sample. This permits modelling n_q as a sequence of uncorrelated samples, $n_q(kT)$, each sample being uniformly distributed. Since $n_q(kT)$ is actually very small in scientific digital simulations, it will be approximated as a weakly stationary zero-mean Gaussian process in the sequel.

For simplicity, $u_a(t)$ and $n_q(kT)$ will be lumped together as a weakly stationary zero-mean Gaussian process $w_I(kT)=w_{I,k}$ with covariance

$$E\{w_{I,k}\, w_{I,j}^T\}=W_I\delta_{kj} \tag{10}$$

The noise effects just discussed are described and analyzed in greater detail in [10].

The dynamics of a general open plant discretization for RTDS can be expressed

$$z_{k+1}=\Phi_I z_k+\Gamma_I u_{f,k}+w_{I,k} \tag{11}$$

$$\hat{y}_k=\hat{C}z_k \tag{12}$$

where $\hat{y}_k$ and $u_{f,k}$ are $\hat{y}(kT)$ and $u_f(kT)$. In order to derive an exact expression for the simulation error e_k, given by

$$e_k=\hat{y}_k-y(kT) \tag{13}$$

adjoin the continuous-time states in (1,4,9) to form

$$\eta^T(t)=[x_f^T(t), x^T(t), x_u^T(t)] \tag{14}$$

with dynamics

$$d\eta(t)=\bar{A}\eta(t)+\bar{B}d\upsilon(t) \tag{15}$$

where

$$\bar{A}=\begin{bmatrix} A_f & 0 & B_f C_u \\ 0 & A & BC_u \\ 0 & 0 & A_u \end{bmatrix} \qquad \bar{B}=\begin{bmatrix} I & 0 \\ 0 & 0 \\ 0 & B_u \end{bmatrix} \tag{16}$$

$$\upsilon^T(t)=[n^T(t),\ w^T(t)] \tag{17}$$

The system (15) has the discretization

$$\eta_{k+1}=\Phi_c\eta_k+w_{c,k} \tag{18}$$

with

$$\Phi_c=\exp\{\bar{A}T\} \tag{19}$$

and where $w_{c,k}$ has the covariance

$$E\{w_{c,k}w_{c,k}^T\}=W_c=\int_o^T \exp\{\bar{A}s\}\bar{B}\Upsilon\bar{B}^T\exp\{\bar{A}^Ts\}ds \tag{20}$$

$$\Upsilon=\text{block diag}\{N,\ W\} \tag{21}$$

The error e_k can now be expressed

$$e_k=\bar{C}\bar{x} \tag{22}$$

$$\bar{x}_{k+1}\ \bar{\Phi}\bar{x}_k+\bar{\theta}_k \tag{23}$$

where

$$\bar{C}=[-\acute{C}\quad 0_{\ell\times n_f}\quad C\quad 0_{\ell\times n_u}] \tag{24}$$

and where $\bar{\Phi}$ is expressed

$$\bar{\Phi}=\begin{bmatrix} \Phi_I & \Gamma_I C_\eta \\ 0 & \phi \end{bmatrix} \tag{25}$$

$$C_\eta=[C_f\quad 0_{m\times n}\quad 0_{m\times n_u}] \tag{26}$$

The noise $\bar{\theta}$ is

$$\bar{\theta}_k^T=[w_{I,k}^T,\ w_{c,k}^T] \tag{27}$$

and has covariance

$$E\{\bar{\theta}_k\ \bar{\theta}_k^T\}=\Theta=\text{block diag}\{W_I,\ W_c\} \tag{28}$$

In [10], two quantities based on e_k are found to be useful in characterizing RTDS fidelty. The average mean-squared error at the sample instants, expressed as

$$e_{ss}^2=\lim_{N\to\infty}\frac{1}{(N+1)}\sum_{k=0}^{N} E\{e_k^T\ e_k\} \tag{29}$$

with $E\{\cdot\}$ denoting expectation, can be used to reflect the response of an RTDS to a steady-state signal with dynamics (3,4). Similarly, the cumulative mean-squared transient error, measured at the sample instants, can be used to describe

$$e_t^2=\lim_{N\to\infty}\sum_{k=0}^{N} E\{e_k^T\ e_k\} \tag{30}$$

where the plant and simulation dynamics consist only of the homogeneous response to a probabilistically distributed initial condition with zero mean, having covariance

$$E\left\{\begin{bmatrix} z_o \\ \eta_o \end{bmatrix}[z_o^T\ \ \eta_o^T]\right\}=\chi_o \tag{31}$$

Total fidelity is then defined by

$$J=e_{ss}^2+\omega e_t^2 \qquad \omega\geq 0 \tag{32}$$

The criterion (32) is natural and intuitively attractive, relating directly to cost functions commonly employed in modern control theory. Moreover, it permits either separate or coordinated consideration of steady-state and transient simulation error.

Calculation of e_{ss}^2 and e_t^2 is extremely simple, given that the plant dynamics (1) and the discretization (11) are asymptotically stable. In this case, $\bar{\Phi}$ in (25) is asymptotically stable and

$$e_{ss}^2=\text{tr}\{P_{ss}\bar{C}^T\bar{C}\} \tag{33}$$

$$e_t^2=\text{tr}\{P_t\bar{C}^T\bar{C}\} \tag{34}$$

where $\text{tr}\{\cdot\}$ denotes the trace operator, and P_{ss} and P_t satisfy the discrete Lyapunov equations

$$\bar{\Phi}P_{ss}\bar{\Phi}^T-P_{ss}+\Theta=0 \tag{35}$$

$$\bar{\Phi}P_t\bar{\Phi}^T-P_t+\chi_o=0 \tag{36}$$

Since (35,36) are linear, J in (32) is given by

$$J=\text{tr}\{P\bar{C}^T\bar{C}\} \tag{37}$$

$$\bar{\Phi}P\bar{\Phi}^T-P+(\theta+\omega X_o)=0 \tag{38}$$

3. Optimal Discretization Formulae

In this Section, the measure J from (32) is exploited as a cost function in deriving an optimal procedure for discretizing plants of the form (1,2) for RTDS. The structure of these discretizations is a generalization of the class of explicit, or predictor-type, linear multi-step integration formulae, which can be expressed

$$\hat{x}_{k+1} = \sum_{i=0}^{\nu} \tilde{\alpha}_i \hat{x}_{k-i} + T\sum_{j=0}^{\zeta} \tilde{\beta}_j \dot{\hat{x}}_{k-j} \qquad \hat{x}\epsilon R^n \tag{39}$$

with $\tilde{\alpha}_i$, $\tilde{\beta}_j$ scalar.

Discrete integration formulae of this form are well suited to RTDS from the point of view of efficiency and simplicity of implementation, since they require only one state derivative evaluation per time step and employ a fixed step size. On the other hand [11, pp. 210-217], and as will shortly be seen, these formulae introduce new dynamic modes not directly related to those of the system being simulated. These "parasitic" modes can, in some cases, be highly oscillatory or unstable, a vivid example of this problem being given in [8]. An important feature of the procedure given below is that asymptotic stability of the continuous-time system (15-17) guarantees that the optimal discretization will be asymptotically stable.

We consider discretization formulae of the form

$$\hat{x}_{k+1} = \sum_{i=0}^{\nu} A_i \hat{x}_{k-i} + T\sum_{j=0}^{\zeta} B_j \dot{\hat{x}}_{k-j} \tag{40}$$

where A_i and B_j have the structure

$$A_i = \begin{bmatrix} \alpha_{i1} & & & \\ & \alpha_{i2} & & 0 \\ & & \ddots & \\ 0 & & & \alpha_{in} \end{bmatrix} \qquad B_j = \begin{bmatrix} \beta_{j1} & & & \\ & \beta_{j2} & & 0 \\ & & \ddots & \\ 0 & & & \beta_{jn} \end{bmatrix} \tag{41}$$

Comparing the coefficients in (39) and (40) from the point of view of design, one notes that the latter formulae have n-times the number of free parameters of the former, permitting closer control of the discretization dynamics. Additionally, this control is achieved with no more arithmetic operations per integration step than in implementing (39).

In order to derive necessary conditions for optimal A_i, B_j in (40, 41), we require a state representation of the discretization. Using (1) and (8, 9), rewrite (40) as

$$\hat{x}_{k+1} = \sum_{i=0}^{\nu} A_i \hat{x}_{k-i} + T\sum_{j=0}^{\zeta} B_j \,[A\,\hat{x}_{k-j} + B\,C_f x_{f,k-j}] \tag{42}$$

where $x_{f,k}$ is the prefilter state at $t = kT$. The prefilter trajectory can be propagated by

$$\begin{bmatrix} x_{f,k+1} \\ x_{u,k+1} \end{bmatrix} = \begin{bmatrix} \Phi_f & \Gamma_{u,f} \\ 0 & \Phi_u \end{bmatrix} \begin{bmatrix} x_{f,k} \\ x_{u,k} \end{bmatrix} + \begin{bmatrix} \bar{n}_k \\ \bar{w}_k \end{bmatrix} \tag{43}$$

which is the exact discretization of the adjoined input and prefilter dynamics:

$$\begin{bmatrix} dx_f \\ dx_u \end{bmatrix} = \begin{bmatrix} A_f & B_f C_u \\ 0 & A_u \end{bmatrix} \begin{bmatrix} x_f \\ x_u \end{bmatrix} dt + \begin{bmatrix} I & 0 \\ 0 & B_u \end{bmatrix} \begin{bmatrix} dn \\ dw \end{bmatrix} \tag{44}$$

Assume, for definiteness, that $\nu \le \zeta$ in (42). This expression, recast in state form, is

$$x_{k+1} = \left[\begin{array}{cccc|cccc|c} q_0 & q_1 & \cdots & q_\zeta & r_0 & r_1 & \cdots & r_\zeta & \\ I & & & & & & & & \\ & I & \ddots & & & & 0 & & 0 \\ & & I & 0 & & & & & \\ \hline & & & & \Phi_f & & & & \Gamma_{u,f} \\ & & & & I & & & & \\ & & & & & I & \ddots & & \\ & 0 & & & & & I & 0 & \\ \hline & & & & & & & & \Phi_u \end{array}\right] x_k + \left[\begin{array}{c} \nu_{I,k} \\ 0 \\ \vdots \\ 0 \\ \hline \bar{n}_k \\ 0 \\ \vdots \\ 0 \\ \hline \bar{w}_k \end{array}\right] \tag{45}$$

where

$$x_k^T = [x_{k}^T,\dots,x_{k-\zeta}^T;\ x_{f,k}^T,\dots,x_{f,k-\zeta}^T;\ x_{u,k}^T] \tag{46}$$

with

$$q_i = \begin{cases} A_i + T\,B_i\,A & i = 0,\ \dots,\ \nu \\ T\,B_i\,A & i = \nu+1,\ \dots,\ \zeta \end{cases} \tag{47}$$

$$r_i = T\,B_i\,B\,C_f \qquad i = 0,\ \dots,\ \zeta \tag{48}$$

A similar expression can be written for the error dynamics (22, 23). In this case we have:

$$\bar{\Phi} = \left[\begin{array}{cccc|cccc|c|c} q_0 & q_1 & \cdots & q_\zeta & r_0 & r_1 & \cdots & r_\zeta & & \\ I & & & & & & & & 0 & 0 \\ & I & & & & & & & & \\ & & \ddots & I\ 0 & & & & & & \\ \hline & & & & \Phi_f & & & & & \Gamma_{u,f} \\ & & & & I & & & & & \\ & & & & & I & & & 0 & \\ & 0 & & & & & \ddots & I\ 0 & & \\ \hline & & & & & & & & \Phi_p & \Gamma_{u,p} \\ \hline & & & & & & & & & \Phi_u \end{array}\right] \tag{49}$$

$$\bar{x}_k^T = [\hat{x}_k^T,\ \dots,\ \hat{x}_{k-\zeta}^T;\ \hat{x}_{f,k}^T,\ \dots\ x_{f,k-\zeta}^T;\ x_k^T;\ x_{u,k}^T] \tag{50}$$

$$\bar{C} = [-\hat{C},\ \ 0_{p\times(\zeta+1)n_f},\ \ C,\ \ 0_{p\times n_u}] \qquad (51)$$

In (49), Φ_p and $\Gamma_{u,p}$ are the plant transition and influence matrices, obtained by an exact discretization of the adjoined plant and input dynamics, similar to (43, 44). If the structure of Θ in (28) is expressed

$$\Theta = \left[\begin{array}{c|ccc} W_I & & 0 & \\ \hline & W_c^{11} & W_c^{12} & W_c^{13} \\ 0 & W_c^{21} & W_c^{22} & W_c^{23} \\ & W_c^{31} & W_c^{32} & W_c^{33} \end{array}\right] \qquad (52)$$

(where the partitioning of W_c corresponds to the state components in (16)), then Θ for (49-51) becomes

$$\Theta = \left[\begin{array}{cc|cc|cc} W_I & 0 & & 0 & & 0 \\ 0 & 0 & & & & \\ \hline & & W_c^{11} & 0 & W_c^{12} & W_c^{13} \\ & 0 & & & & \\ & & 0 & 0 & 0 & 0 \\ \hline & & W_c^{21} & 0 & W_c^{22} & W_c^{23} \\ & 0 & & & & \\ & & W_c^{31} & 0 & W_c^{32} & W_c^{33} \end{array}\right]$$

The A_i and B_j in (47, 48) can be isolated by rewriting $\bar{\Phi}$ as

$$\bar{\Phi} = \phi + S\ K\ \Lambda \qquad (53)$$

with

$$K = [A_0,\ A_1,\ \dots,\ A_\nu;\ B_0,\ B_1,\ \dots,\ B_\zeta] \qquad (54)$$

The matrices comprising (54) are dimensioned as follows:

$$\left.\begin{array}{l} \phi \in R^{n_\ell \times n_\ell} \quad S \in R^{n_\ell \times n} \quad K \in R^{n\times m_\ell} \quad \Lambda \in R^{m_\ell \times n_\ell} \\ n\ = (n+n_f)\ (\zeta+1) + n + n_u \qquad m_\ell = n(\nu+\zeta+2) \end{array}\right\} \qquad (56)$$

and have the structure

$$\phi = \left[\begin{array}{cccccc|ccccc|c|c} 0 & & & & & & & & & & & & \\ I & & & & & & & & & & & & \\ & I & & & & & & & & & & & \\ & & \cdot & & & & & & 0 & & & 0 & 0 \\ & & & \cdot & & & & & & & & & \\ & & & & \cdot\ I & 0 & & & & & & & \\ \hline & & & & & & \Phi_f & & & & & & \Gamma_{u,f} \\ & & & & & & I & & & & & & \\ & & & & & & & I & & & & 0 & \\ & & & & & & & & \cdot & & & & \\ & & & & & & & & & \cdot & & & \\ & & & & & & & & & \cdot\ I & 0 & & \\ \hline & & 0 & & & & & & & & & \Phi_p & \Gamma_{u,p} \\ \hline & & & & & & & & & & & & \Phi_u \end{array}\right] \qquad (57)$$

$$S^T = [I_n \quad 0_{n\times(n_\ell - n)}] \qquad (58)$$

$$\Lambda = \left[\begin{array}{cccccc|cccc|c} I_n & & & & & & & & & & \\ & I_n & & & & & & 0 & & & \\ & & \cdot & & & & \nu + 1 & \text{entries} & & & \\ & & & \cdot & & & & 0 & & & \\ & & & & \cdot\ I_n & & & & & & \\ \hline TA & & & & & & TBC_f & & & & 0_{m_\ell\times(n+n_u)} \\ & TA & & & & & & TBC_f & & & \\ & & TA & & & & & & \cdot & & \\ & & & \cdot & & & & & & \cdot & \\ & & & & \cdot & & & & & & \\ & & & & & \cdot\ TA & & & & \cdot\ TBC_f & \end{array}\right]$$

$$\underbrace{\qquad\qquad}_{\zeta\ +\ 1\ \text{entries}} \qquad (59)$$

The free parameters in K can be isolated by reexpressing (54) in the form

$$K(\gamma) = \Psi[I_{m_\ell} \otimes \gamma^T]\Omega \qquad (60)$$

where $\otimes$ denotes the Kronecker product. The constant factors in (60) are

$$\Psi = [I_n^{(1)} \ \vdots\ I_n^{(2)} \ \vdots\ \cdots \ \vdots\ I_n^{(m\ell/n)}] \qquad (61)$$

(having $m\ell/n = \nu+\zeta+2$ I_n entries) and

$$\Omega = [E_1 \ \vdots\ E_2 \ \vdots\ \cdots \ \vdots\ E_m\ ]^T \qquad E_i = e_i\ e_i^T \quad (62)$$

where e_i is the i^{th} standard basis vector of $R^{m\ell}$. The free parameters, γ, are now in the form

$$\gamma^T = [\alpha_{01}, \alpha_{02},\ \dots,\ \alpha_{0n}, \alpha_{11},\ \dots, \alpha_{\nu n}; \beta_{01},\ \dots, \beta_{\zeta n}] \qquad (63)$$

From (37) and (38), optimal γ will minimize the Lagrangian

$$L(P,L,\gamma)+\text{tr}\{P\ \bar{C}^T\bar{C}\}+\text{tr}\{[\bar{\Phi}P\bar{\Phi}^T-P+(\Theta+\alpha X_o)]L^T\} \qquad (64)$$

where L is a matrix of Lagrange multipliers and X_o is the covariance of $\bar{x}_o$ in (31). The necessary conditions for optimality are

$$\frac{\partial L}{\partial P} = 0 \qquad \frac{\partial L}{\partial L} = 0 \qquad \frac{\partial L}{\partial \gamma} = 0 \qquad (65)$$

Expanding the first two equations in (65) gives

$$\frac{\partial L}{\partial P} = \bar{\Phi}^T L \bar{\Phi} - L + (\Theta+\alpha X_o) = 0 \qquad (66)$$

$$\frac{\partial L}{\partial L} = \bar{\Phi} P \bar{\Phi}^T - P + \bar{C}^T\bar{C} = 0 \qquad (67)$$

After some manipulation (deferred to the Appendix), the m_ℓ elements of the third necessary condition take the form

$$\left(\frac{\partial L}{\partial \gamma}\right)_i = e_i^T\ [2\Lambda L N_1 + 2\ N_2 \text{diag}\{\gamma\}\ \Lambda L \Lambda^T]e_i = 0$$
$$i = 1,\ \dots,\ m_\ell \qquad (68)$$

Thus, in (68), the elements in the gradient are merely the entries on the main diagonal of the RHS. The matrices N_1 and N_2 are

$$N_1 = \phi \; P \; S \; \Psi \tag{69}$$

$$N_2 = \Psi^T \; S^T \; P \; S \; \Psi \tag{70}$$

A simple numerical algorithm for computing optimal γ is given below:

0. Choose any value for γ which results in a stable discretization ($\gamma = 0$ is acceptable). Set $k = 0$.

1. Solve (66, 67) for L_k, P_k.

2. Calculate $\partial L/\partial\gamma_k$ from (68)

3. Set

$$\gamma_{k+1} = \gamma_k - \xi \; \partial L/\partial\gamma_k \tag{71}$$

where $\xi \in (0, 1]$ is chosen to ensure that

$$J_{k+1} < J_k = \mathrm{tr}\{P_k \; \bar{C}^T\bar{C}\} \tag{72}$$

4. Set $k = k+1$ and go to 1.

This is a steepest descent-type adaptation of a globally convergent algorithm described in [12-14] for the optimal output feedback regulator problem. Because of the potentially large dimensionality in the dynamics (23), it is worthwhile to note that this need not present difficulties during the solution of the Lyapunov equations in Step 3. Numerical stability can be enhanced by exploiting the block-upper triangular structure of Φ in (49) to decompose the calculation of L_k, P_k into lower dimensional subproblems. Further, the overall computational burden can be significantly eased by exploiting the fact that the lower right-hand block of Φ, L_k and P_k (reflecting the prefilter, plant and input exact descretizations) do not change during execution of the algorithm.

4. Conclusions

A scalar figure of merit has been derived for characterizing the fidelty of a generic RTDS, reflecting the full end-to-end dynamics of the simulator. This figure of merit has been applied to the problem of optimally discretizing continuous-time plant dynamics for RTDS. The class of discretization considered was an efficient generalized form of linear multistep discrete integration formulae. Necessary conditions for optimality with respect to the fidelity measure, and a simple numerical design algorithm were derived. An application to simulating the lateral dynamics of a commercial jet transport is under development.

ACKNOWLEDGEMENT: This work was supported by NASA Langley Research Center under Contract NAS1-17161.

REFERENCES

1. Martens, H., "A Comparitive Study of Digital Integration Methods," Simulation, Feb. 1969, pp. 87-94.

2. Baron, S., Muralidharan, R., and Kleinman, D., "Closed-Loop Models for Analyzing Engineering Requirements for Simulators," NASA CR-2965, Feb. 1980.

3. Sage, A., and Smith, S., "Real-Time Digital Simulation for Systems Control," Proc. IEEE, Vol. 54, No. 12, Dec. 1966, pp. 1802-1812.

4. Fowler, M., "A New Numerical Method for Simulation," Simulation, Vol. 6, Feb. 1966, pp. 90-92.

5. Smith, J., Mathematical Modeling and Digital Simulation for Engineers and Scientists, John Wiley and Sons, New York, NY, 1977.

6. Nigro, B., "An Investigation of Optimally Stable Numerical Integration Methods with Application to Real-Time Simulation," Simulation, Nov. 1969, pp. 253-264.

7. Barker, L., Bowles, R., and Williams, L., "Development and Application of a Local Linearization Algorithm for Integration of Quaternion Rate Equations in Real-Time Flight Simulation Problems," NASA TN-D-7347, Dec. 1973.

8. Rolston, D., "Sinusoidal Integration for Simulation of Second-Order Systems," Proc. AIAA Flight Simulation Tech. Conf., Niagra Falls, NY, June 13-15, 1983.

9. Halyo, N., and Stallman, S., "A Parametric Study of Aliasing Error for a Narrow Field of View Scanning Radiometer," NASA CR-3294, Aug. 1980.

10. Halyo, N., Moerder, D., and Broussard, J., "Measures of End-to-End Fidelity for Digital Real-Time Simulations," proposed NASA CR.

11. Henrici, P., Discrete Variable Methods in Ordinary Differential Equations, John Wiley and Sons, New York, NY 1962.

12. Halyo, N., and Broussard, J., "A Convergent Algorithm for the Stochastic Infinite-Time Discrete Optimal Output Feedback Problem," Proc. JACC, Charlottesville, VA, 1981.

13. Moerder, D., and Calise, A., "Convergence of a Numerical Algorithm for Calculating Optimal Output Feedback Gains," to appear in IEEE Trans. Auto. Control.

14. Makila, P., "On the Anderson-Moore Method for Solving the Optimal Output Feedback Problem," IEEE Trans. Auto. Control, Vol. AC-29, Sept. 1984, pp. 834-836.

Appendix

For convenience, isolate that portion of L in (64) which is an explicit function of γ:

$$L = L_o + \tilde{L}(\gamma) \tag{A.1}$$

$$\tilde{L}(\gamma)=\mathrm{tr}\{[2\phi PSK(\gamma)\Lambda+\Lambda^T K^T(\gamma)S^T PSK(\gamma)\Lambda]L\} \tag{A.2}$$

We now derive the form of the individual elements of $\partial L/\partial\gamma$ as given by (68). Note, from (61) and (62), that

$$[I_{m\ell} \otimes \gamma^T]\Omega = \mathrm{diag}\{\gamma\} \tag{A.3}$$

In order to demonstrate this, recall a standard identity for Kronecker products, which states that

$$(A \otimes B)(C \otimes D) = (AC \otimes BD) \tag{A.4}$$

when all products are meaningful. Now, $I_{m\ell}$ and Ω can be expressed as

$$I_{m_\ell} = \sum_{i=1}^{m\ell} e_i\, e_i^T \qquad \Omega = \sum_{j=1}^{m\ell} e_j \otimes e_j\, e_j^T \tag{A.5}$$

Substituting (A.5) into (A.3), using (A.4) and the fact that $e_i^T e_j = \delta_{ij}$, we obtain

$$[I_{m_\ell} \otimes \gamma^T] = \sum_{i=1}^{m_\ell} \gamma_i\, e_i\, e_i^T \tag{A.6}$$

which is the RHS of (A.3).

Examining the first term of (A.2), define

$$b_j^{(1)}=\frac{\partial}{\partial\gamma_j}\mathrm{tr}\{2\phi PSK(\gamma)\Lambda L\}=\mathrm{tr}\{2\phi PS\Psi\; e_j e_j^T\Lambda L\} \tag{A.7}$$

which can be reexpressed

$$b_j^{(1)}=2\; e_j^T\Lambda L\phi PS\Psi e_j=2\; e_j^T\Lambda LN_1 e_j \tag{A.8}$$

In the second term of (A.2), differentiate by parts to define

$$b_j^{(2)}=\frac{\partial}{\partial\gamma_j}\mathrm{tr}\{\Lambda^T K^T(\gamma)S^T PSK(\gamma)\Lambda L\} \tag{A.9}$$

$$=\mathrm{tr}\{\Lambda^T e_j e_j^T \Psi^T S^T PSK(\gamma)\Lambda L+\Lambda^T K^T(\gamma)S^T PS\Psi e_j e_j^T\Lambda L\} \tag{A.10}$$

$$=\mathrm{tr}\{\Lambda^T e_j e_j^T N_2\; \mathrm{diag}\{\gamma\}\; \Lambda L+\Lambda^T\; \mathrm{diag}\{\gamma\}\; N_2 e_j e_j^T \Lambda L\} \tag{A.11}$$

by (69) and (A.3). Using invariance of the trace under commutation and the symmetry of N_2, (A.11) becomes

$$b_j^{(2)} = 2\; e_j^T\; N_1\; \mathrm{diag}\{\gamma\}\; \Lambda L\Lambda^T\; e_j \tag{A.12}$$

The j^{th} element of $\partial L/\partial\gamma$, then, is (68); the sum of (A.7) and (A.12).

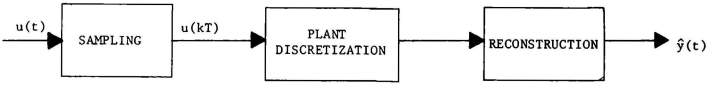


FIGURE 1. IDEALIZED RTDS

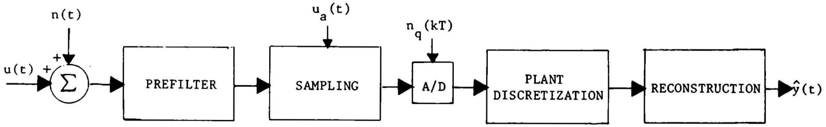


FIGURE 2. CAUSALITY RELATIONS IN A GENERIC RTDS

85-1741

FLIGHT SIMULATION FIDELITY IN A TOTAL G-FORCE ENVIRONMENT

David E. Bischoff* and David A. Poole**
Veda Incorporated
Warminster, Pennsylvania
and
Jacob Eyth, Jr.***
Naval Air Development Center
Warminster, Pennsylvania

Abstract

An evaluation of the Dynamic Flight Simulator (DFS) installed on the NAVAIRDEVCEN human centrifuge was conducted to determine the capability of the DFS to simulate transient and steady state aircraft force and motion characteristics in a total G-force environment. The evaluation was performed in both fixed and moving base modes utilizing standard qualitative and quantitative flight test techniques. The DFS demonstrated exceptional visual and motion cues for out-of-control/spin conditions as well as the potential to be an effective research tool for a variety of high-G maneuvering tasks. High fidelity with F-14A airplane high angle of attack/spin conditions was also demonstrated. Continued development of the DFS is necessary to optimize dynamic response of the centrifuge for specific high G maneuvering tasks.

Nomenclature

A	-	Centrifuge Roll Gimble Angle (deg)
A/C	-	Simulated Aircraft
B	-	Centrifuge Pitch Gimble Angle (deg)
C/F	-	Centrifuge
G	-	Total acceleration - (g's)
G_R	-	Radial acceleration - (g's)
G_T	-	Tangential acceleration - (g's)
G_X	-	Longitudinal acceleration (g's)
G_Z	-	Vertical acceleration - (g's)
K	-	Gain constant
p	-	Roll rate - (deg/sec)
q	-	Pitch rate - (deg/sec)
r	-	Yaw rate - (deg/sec)
R	-	Centrifuge arm radius - (50 ft)
s	-	LaPlace operator
ω	-	Centrifuge arm rate - (rps)
τ	-	Time constant - (sec)
$(\hat{\ })$	-	Estimated value
$(\dot{\ })$	-	Time derivative

Introduction

Background

Flight simulations of tactical aircraft have generally lacked realism in the maneuvering environment. The high forces characteristic of tactical maneuvering are difficult to simulate with conventional motion base schemes. Physical limits on the allowable displacements of the simulator restrict the magnitude and duration of the forces used for maneuvering cues. Motions of the simulator must be attenuated, usually with high-pass filters, which unfortunately introduce unacceptable phase lag and attenuation characteristics. Since accelerations cannot be sustained, high or sustained maneuvering forces must be simulated using g-suit, g-seat, helmet/arm loader, or light dimming systems. The inability to produce representative force cues detracts from the realism of the simulation, particularly for air combat maneuvering and out-of-control flight phases.

The Naval Air Development Center (NAVAIRDEVCEN) has developed a total G-force Dynamic Flight Simulator (DFS). Using the NAVAIRDEVCEN's human centrifuge as a motion and force base, the DFS is designed to be capable of reproducing the total multi-directional G-force environment associated with flight of modern high performance aircraft in response to pilot control inputs. As such, the DFS is designed to be used as a safe platform for evaluating new concepts in crew-station design, cockpit displays and controls, restraint systems, aerodynamic configurations and handling qualities as well as conducting training in pilot procedures in the G-environment in which they are designed to be utilized.

A number of previous aircraft simulation programs have been conducted on the NAVAIRDEVCEN centrifuge. These included simulation of B-720 Severe Air Turbulence, F-4 and F-14 spins, F-4 buffet during sustained G-profiles, and A-7 night catapult launching programs. While these programs demonstrated the potential of the centrifuge for flight simulation of steady state G-profiles, they also pointed out the disorienting angular artifacts imposed upon the pilot during transient G-vector rotations, which at the time limited the usefulness of the centrifuge as a true flight simulator. A NAVAIRDEVCEN Independent Research/Independent Exploratory Development program[1] has since identified methods of controlling these angular discrepancies, thus paving the way for development of the DFS.

The initial program performed on the DFS was a study of the departure/spin characteristics of the F-14 aircraft[2]. The F-14 has been shown[3] to exhibit a flat spin mode with yaw rates and longitudinal accelerations in excess of 150 deg/sec and negative 5G's (eyeballs out), respectively, at the pilot's station. Such conditions are extremely difficult to safely investigate in the real aircraft but are well suited to centrifuge simulation.

* Aerospace Engineer, Member AIAA
** Engineering Test Pilot
*** Program Manager, Member AIAA

Prior to initiating the actual F-14 spin studies and making the DFS available for other programs, it became apparent that a definition of the DFS' capabilities was required. A program to validate the DFS' flight characteristics was therefore developed by the NAVAIRDEVCEN and conducted with pilot support from Veda Incorporated.

Purpose

The purpose of this program was to 1) define the operational capabilities of the DFS to simulate both transient and steady state aircraft force and motion characteristics during controlled and out of control flight scenarios and 2) to validate the handling qualities of the DFS with respect to the F-14 aircraft.

This paper discusses the integration of the various DFS equipments into a total simulation facility, presents the resultant flight fidelity characteristics and projects future applications of the DFS.

Description of DFS

The DFS was designed and built to take advantage of the unique force and motion generating capabilities of the NAVAIRDEVCEN's three-degree-of-freedom man-rated centrifuge. The centrifuge includes a 10-foot diameter gondola whose angular rotation is controlled through a two-gimbal drive system and is mounted at the end of a 50-foot long arm. This system, by itself, is capable of attaining a 10 G/sec onset rate between 1.5 and 15 G's and accelerating a typical one ton cockpit plus subject and peripheral equipment to 15 G's.

The DFS, schematically presented in figure 1, was developed by inserting a reconfigurable cockpit (currently implemented to represent the F-14 aircraft) within the centrifuge's gondola and connecting the system to the Center's CDC Cyber 170/760 digital computer system, via a fiber optic link, to provide the necessary real-time computational requirements[4]. Control of the centrifuge is accomplished via an EAI 231R analog computer.

The crewstation installed in the gondola consists of a multipurpose cockpit, a removable instrument panel display, and a real world Redifusion SP-2 visual display unit. The cockpit's variable geometry design gives it the flexibility of modelling various single-place cockpit configurations, as does the removable instrument panel. Information is displayed to the pilot via simulated aircraft instruments and programmable Collins color cockpit displays. Pilot control of the DFS is accomplished via a McFadden, hydraulic, three-axis (stick and rudder) control loader system and an F-14 throttle assembly.

The DFS can be operated in a number of modes ranging from fixed-base to fully dynamic. With the cockpit removed from the gondola, it can be setup as a purely fixed-base simulator with all simulated aircraft and visual display systems operating. A static or fixed base mode is also available with the cockpit installed in the gondola prior to energizing the centrifuge. A limited motion, or gimbals-only mode can be achieved by energizing all but the centrifuge arm. This mode provides limited pitch and roll motions of the simulated aircraft but without the accompanying G-profiles. Finally, a fully dynamic mode is available by energizing all of the centrifuge systems.

The DFS control software includes an F-14 aerodynamic data package, a nonlinear flight control system capability, various trim capabilities and control algorithms for the centrifuge drive, the instrumentation, and the visual display system. The software generates all aircraft forces and moments using non-linearized equations of motion and three-dimensional data storage[5].

The aerodata package is capable of being changed from one aircraft to another without major reprogramming and to input assymmetries in mass, engine, or aerodynamic features as required by the experiment scenario. It has no singularities in its earth referenced attitude angles, and permits large angular excursions in both angle-of-attack and sideslip.

Scope of Tests

The scope of the validation effort was established by the available F-14 math model. Only the clean (no stores) loading was evaluated in cruise (landing gear and flaps UP) configurations. Tests were conducted with the Stability Augmentation System (SAS) either ON or OFF, as applicable. Dynamic operational limits imposed by structural, safety and physiological considerations are presented in Table 1.

The real-time validation effort was performed primarily by two experienced high angle of attack flying qualities test pilots from Veda Incorporated, Lexington Park, MD over a period of one year. A total of eight days of fixed base operations, for approximately 34 hours, and 16 separate dynamic operations, totaling 18.2 hours, were performed by these pilots. In addition, comments and suggestions concerning the fidelity of the DFS with respect to the F-14 were obtained from an additional 10 fleet and test pilots who participated in the spin experiments. Selected comments from all the pilot's who have flown the F-14 DFS are presented in figure 2.

Method of Tests

Tests to determine the fidelity of the DFS with respect to the F-14 aircraft were conducted in both batch and real-time simulation modes. The batch mode was used to calibrate control system components, verify weight and center of gravity routines, analyze steady state trim conditions and provide preliminary indications of time history response characteristics to simulated control inputs. All final fidelity tests were conducted in the real-time mode in order to obtain qualitative pilot opinions of simulated responses as well as quantitative data. Standardized flight test techniques were utilized to determine the fidelity of the DFS. Where possible, DFS data were quantitatively evaluated against F-14A published flight test data.

Test instrumentation consisted of simulated cockpit instruments, tape measure, hand-held force gauge, stopwatch and three 8-channel strip chart recorders. Data were also displayed digitally on the Project Officers Display and on the Data Display Terminal in the Experiment Control Station. All dynamic centrifuge operations were conducted under the guidance of a biomedical team, who maintained visual and voice communication with the pilot and continually monitored his heart-beat, pulse rate and blood pressure.

Results and Discussion

Control Loader Implementation

The cockpit control force loader is a McFadden hydraulic system which was modified specifically for the DFS application. The nonlinear characteristics of the McFadden system were removed and the electronics repackaged for mounting within the centrifuge's gondola. The control loader provided very good control force characteristics. There were, however, some unique DFS operational characteristics which had to be dealt with. Rotation of the gondola from an upright position resulted in movement of the control stick from its trim position due to the varying orientation of the G-Vector. This movement was characterized by the pilots as a stick float which resulted in a degradation of closed-loop handling qualities. Special analog computer circuitry was developed to provide an external force command to the stick in response to measured centrifuge accelerations, thereby restraining it at its trim condition. In addition, the analog computer was used to provide a simulated bobweight force in response to computed aircraft G_Z. This bobweight term enhanced the realism of the simulation and improved the pilot's controllability.

The DFS' control loader maximum stick position limits result in a rectangular pattern of maximum stick deflections defined by fore, aft, left and right limits. Since the F-14 uses a common control surface for longitudinal and lateral control, combined maximum pitch and roll commands results in reduced surface deflections. The DFS's surface deflections are limited to coincide with those of the aircraft. These reduced surface deflections are reflected in the cockpit stick position as indicated in figure 3. The McFadden control loader does not have the capability to simulate these nonlinear control deflection limits. Therefore it was necessary to implement them via the external force command signal from the 231R analog computer. This implementation proved very effective and provided the pilots with a positive indication of the proper stick location for maximum lateral stick deflection during spin entries and recoveries.

Fixed Base Validation

The piloted evaluation was initiated with the pilots performing standard flying qualities tests; qualitative evaluations, doublets and step inputs. It was obvious from the outset that excessive lags were present in the visual and cockpit displays, resulting in the inability of the pilots to precisely control attitude and accomplish closed-loop tasks. These delays were measured by using a video recorder to record both the cockpit control inputs, displayed on a strip chart recorder, and the VDI attitude responses, displayed on a CRT repeater located alongside the strip chart parameters. Delays of as much as 300 msec were determined to be present between the initial control stick movement and the resulting attitude response. Three contributors to this delay were uncovered through further analysis: a 50 msec (one frame time) delay in the simulated control surface actuator response, inherent delays in the cockpit display units, and the software calling heirarchy.

The cockpit display driver manufacturer was alerted to this problem. He was able to increase the refresh rates and streamline the display driver software to reduce the inherent display delay to 80 - 100 msec.

The Simulation Control Computer (SCC) software hierarchy utilizes serial processing to and from the DFS cockpit, the Central Computer System and the project engineer's control and display stations. The input/output processing and the computational tasks necessary to implement the simulation program result in a computer frame time of 50 msec as shown in figure 4. At the beginning of the evaluation, the SCC placed priority on the information being sent to the operator's control consoles and displays rather than to the pilot's cockpit displays. In addition, this information had actually been calculated in the previous computer frame time due to the SCC's calling structure, contributing significantly to the lags and resulting control instabilities noted by the evaluation pilots. The SCC input/output processing priorities were modified to complete the entire process within a single frame, as shown in figure 4, resulting in greatly improved system response characteristics.

The control surface actuators were modelled in the DFS software as outputs of a Tustin transform filter which was preprocessed and stored in the computer program via a state space representation. For time constants of 50 msec or greater, a one frame time pure delay of 50 msec resulted prior to initiation of the exponential output of the simulated filter. For time constants less than 50 msec, there was no delay in the response, however, the output of the filter increased linearly and oscillated about the commanded level. Neither of these conditions provided a satisfactory response and contributed significantly to the pilot's closed-loop control problems noted. Since the Euler integration method utilized in the program essentially resulted in a three frame time delay from the acceleration command to the attitude output, which was then compounded by the filter modelling errors, it was decided to eliminate the dynamic actuator modelling and drive the control surfaces directly from the cockpit commands.

As a result of these modifications, the system delays were reduced to approximately 100 - 140 msec, resulting in satisfactory handling qualities during maneuvering tasks.

Moving Base Validation

General The motion base was utilized in two modes with the DFS. The first of these was the gimbals-only mode. This mode was essentially a two degree-of-freedom motion base. Pitch rotation and roll were the only allowable motions. The fully dynamic mode was essentially a three degree-of-freedom motion base, a dual-gimballed centrifuge.

Continued DFS development was underway throughout the moving base evaluations. Roughly 18 hours were spent evaluating the fidelity of the spin simulation, and troubleshooting the control algorithm. For most test conditions, lightly damped structural oscillations were easily excited by rapid control inputs, most notably during G reduction, as in the release of the stick force following a pull up recovery, or as in a push over from the platform G level. The resulting oscillations in pitch and roll (which could be excited by inputs in either control axis) led to a closed-loop instability (PIO) that was sometimes so violent that the stick could not be held fixed. Filtering the acceleration input to the centrifuge drive proved effective in eliminating these oscillations.

Physiological Effects Some rather unique sensations were experienced as the result of the force generated in the spinning centrifuge. The centrifuge simulates the 1G aircraft condition by operating about a platform level of 1.55G, as shown in figure 5. With the centrifuge rotating to produce this "simulated" 1 G at the pilot's station, any movement of the head, whether it be rotation (side-to-side) or tilt (roll or pitch), induced a rolling sensation. Such head movements caused the eye to "see" the entire simulator roll. Combinations of these simple head movements produced additive effects (e.g. left rotation and down tilt combined to induce a severe left rolling sensation). This effect was intensified as the centrifuge's rotation rate was increased to increase the G level. The strength of this sensation was extremely pilot dependent. All the pilots who flew the DFS noticed it; one was unaffected by it, some emphatically objected to it, while most learned to adjust to or tolerate it.

In order to minimize the effect of these sensations, head movements were discouraged during centrifuge runs. This limitation on movements of the pilot's head will make the DFS unsuitable for flight phases like air combat maneuvering and formation flying, although certain tasks within these broad flight phases can be satisfactorily simulated.

Structural Modes Operating the simulator at the end of a 50 foot arm gave ample opportunity to observe some lightly-damped structural modes. Stepping onto the gondola, a 2-3 Hz oscillation could be easily excited. This was primarily an oscillation of the outer (roll or A) gimbal. A slightly higher frequency "bounce" mode, identified with the centrifuge arm, was sometimes superimposed on the roll gimbal mode. The outer gimbal drive appeared to have no freeplay, but the combined torsions of the gimbal drive shafts provided enough amplitude to make the oscillations quite noticeable and, at times, objectionable. A similar inner (pitch or B) gimbal mode was accentuated by freeplay in the split drive to the two B gimbal drives. This freeplay caused the excitation of the B gimbal response mode whenever the B gimbal was driven at a realistic aircraft frequency. Oscillations resulting from the excitation of these lightly-damped structural modes were dubbed the "hobbyhorse" modes. Coupling of these modes with the system through the combined inertia of the pilots arm and the control stick often led to control difficulties in early configurations of the centrifuge control algorithm.

Centrifuge Control Algorithm Optimization During this phase of the DFS Validation effort, it became apparent that some fine tuning of the new Centrifuge Control Algorithm (CCA), which was now for the first time coupled with a visual display, could improve the pilot's perceived response. At the same time, cross axis coupling and mechanical slip in the gimbal actuators were identified as detracting from the pilot's acceptance of the DFS motion. The mechanical slip was reduced by tightening the gimbal drive linkages. The angular motion attenuation and aircraft rate lead terms in the CCA were "optimized" in a trial and error process to achieve acceptable pilot perceived responses and a high order filter added at the gimbal command outputs to separate the remaining unwanted mechanical response of the centrifuge from the commanded aircraft responses.

This configuration resulted in improved cockpit motions in which the pilot was able to vigorously move the stick ±0.5 to ±1.0 inches before excessive cross axis coupling and/or hobby-horsing were encountered. Finally, it was decided to remove the aircraft G_Y term and filter the total G command used to drive the centrifuge arm. This was done in order to reduce the transient G commands being sent to the arm rate command. This final configuration, presented in figure 6 and discussed in reference 1, was very successful in reducing the hobby-horse motion excited during G-transients.

The Centrifuge Control Algorithm (CCA) will require the implementation of complex filters in order to separate the aircraft response modes (.7 Hz) from the DFS centrifuge gimbal mechanical response ("hobby-horse") modes. The overall design goal should be to achieve DFS phasing coherency at all the input frequencies required. In order to achieve this goal, the CCA will probably require implementing in either digital or hybrid form to allow complex digital filtering techniques to be applied. The main filters will be required in the B and A gimbal drive outputs and should provide steep low-pass characteristics, with a roll-off of around 1 Hz with minimum phase shift. In addition, it is probable that a first order filter will be required in the G_Z input to smooth overall centrifuge response.

Gimbals-Only Mode Two short evaluation periods early in the tests were devoted to the gimbals-only mode. This mode was operated with the gimbal drives energized in their normal CCA implementation while the centrifuge arm was maintained stationary. Since the control algorithm was not developed for such operation, this mode was not optimized. However, the gimbal motions, coupled with the instruments and visual displays, provided an extra measure of realism beyond that available from the fixed base simulation, contributing positively to the pilot's perception of angular response. With a proper control algorithm, the gimbals-only mode shows potential for various applications including DFS familiarization and build-up tests, as well as "near 1 G" flying qualities work in general. This mode would also be useful in investigating structural modes and in evaluating potential damping schemes.

Fully Dynamic Mode During the initial rotational acceleration to the 1.55 G "platform" there was a strong sensation of rolling and yawing to the left (the direction of centrifuge rotation). This sensation rapidly (but smoothly) subsided as the centrifuge steadied at the required "platform" rotation rate. Only a very short time (less than one minute) was necessary to adapt to the platform state. The platform state "felt" like normal 1 G flight.

Transfer of control of the centrifuge from the normal electromechanical control to digital computer control, and then to pilot control was smooth and easy. A small, minor bump might be felt but no significant transients or gross out-of-trim conditions occurred during these tests.

The stopping sequence normally involved re-establishment of the "platform" G level, followed by a smooth and gradual deceleration to a full stop in the upright attitude. Throughout these tests the operation of the stopping sequence was comfortable.

With the latest implementation of the control algorithm[1], the handling qualities of the DFS were impressive. In tests of the F-14 up to approximately 350 KIAS, the controls could be handled as roughly as desired without any adverse characteristics. Sinusoidal stick pumping to approximately 10 rad/sec uncovered no "sensitive" frequencies. The response of the DFS was smooth, predictable, and very much "just like an airplane."

During high G turns and pull ups, the sustained maneuvering forces made the simulation extremely realistic. There was no noticeable difference in the feel of left and right turns. Forces during aerobatics and tactical maneuvering added significantly to the realism of the simulation. The realistic maneuvering forces are an enhancing characteristic.

A first order filter on the centrifuge acceleration command was implemented effectively blocking excitation of the gimbal mode oscillations. Unfortunately, there was a tradeoff with the overall Nz response as a function of the filter time constant as shown in figure 7. A one second time constant caused a noticeable lag in the perceived G response to pitch commands. The B gimbal response to pitch commands was immediate, however, providing a strong feedback cue to the pilot. Similarly, the visual system, VDI and the pilot's G-suit pressurization control were unaffected by the G lag, providing a strong feedback to the pilot for low airspeed points. Although the G lag could be noticed, the pitch rate and visual cues essentially covered it up. At higher speeds (above about 350 KIAS), however, the nearly one second lag was excessive and a tendency for a low frequency pilot-induced-oscillation was evident. At time constants of 0.5 seconds or less, the G lag was reduced however the gimbal oscillations were considered excessive. A compromise was established at a time constant of 0.75 seconds to reduce both the G lag and the gimbal oscillations. Though the particular filter time constant chosen worked only in a limited bandwidth, the ability to virtually eliminate the "hobbyhorse" oscillations was clearly demonstrated. Advanced filtering techniques should be able to extend the bandwidth within which the DFS can be operated free from structural oscillations.

High Angle-of-Attack Response Handling qualities at high angles of attack (above 25 units) in 1 G and accelerated flight at airspeeds less than 0.6 M were good throughout the tests. The low G available at these low airspeed test conditions effectively made no large demands on the centrifuge drive. As expected, adverse coupling or structural mode excitations were not a problem at these flight conditions. Objective fidelity of DFS characteristics with F-14 airplane flight data was demonstrated as shown in figure 8. The excellent high angle-of-attack flying qualities enhance the realism of the DFS F-14A simulation.

Spins The DFS provided an extremely realistic simulation of the cockpit environment during out-of-controlled flight. The forces during departures were especially realistic, and the motion sensations during the early stages of the spin were highly convincing. All equipment in the DFS operated normally in repeated spins out to 5 G longitudinally. All instruments gave correct indications, reinforcing the spin cues. High objective fidelity with actual F-14A spin characteristics was demonstrated. A typical spin simulation run is presented in figure 9. The DFS is an exceptional device for spin simulations and has excellent potential for a wide variety of out-of-controlled flight applications.

Summary

The DFS provides excellent steady state G-force and maneuvering cues. It shows potential to be an effective research tool and flight simulator for a wide variety of maneuvering tasks and projects. Upgrading various components of the DFS will open up other potential areas where the unique capabilities of the DFS can be exploited. Certain tasks were not well-suited to the DFS; notably any task requiring the pilot to move his head around the cockpit. A summary of current and projected DFS capabilities is presented in Table 2.

References

1. Crosbie, R. J. and Kiefer, D. A., "Controlling the Human Centrifuge as a Force and Motion Platform for the Dynamic Flight Simulator." AIAA Paper 85-1742, Flight Simulation Conference, July 1985.

2. Eyth, J. and Gleisner, D., Lt USN, "Application of the Dynamic Flight Simulator (DFS) to Evaluate Pilot Performance in a Simulated F-14 Flat Spin Environment" AIAA Paper 85-1730, Flight Simulation Conference, July 1985.

3. Weber, LT C. L., et al, "Final Report, Model F-14A Airplane High Angle of Attack Flying Qualities Evaluation," NAVAIRTESTCEN Report SA-C11R-77, 20 Dec 1977.

4. Crosbie, R. J., Eyth, J., Thomas, G. T., et al, "NAVAIRDEVCEN Dynamic Flight Simulator F-14 Spin Simulation Program System Description and Specification Report," NADC-81145-60, April 1981.

TABLE 1 - DFS CENTRIFUGE LIMITATIONS

Longitudinal Acceleration - G_X	± 5G
Lateral Acceleration - G_Y	± 1G
Normal Acceleration - G_Z	+ 1,+5G
Acceleration Onset Rate	± 2.5 G/sec
Maximum Gimbal Rate	± 30 deg/sec

TABLE 2 - DFS CAPABILITIES

AREA OF INTEREST	DFS UTILITY		
	ESPECIALLY USEFUL	GOOD POTENTIAL	NOT SUITABLE
High AOA	●		
Spins	●		
Departures	●		
Restraint Systems	●		
High g seats	●		
Physical Workload	●		
High g onset rates		●	
Pilot Performance		●	
General Flying Qualities		●	
IFR		●	
Takeoff/Landing		●	
Low AOA		●	
Beyond $C_{L_{max}}$		●	
Direct Force Control		●	
Terrain Following		●	
Tracking Tasks		●	
Weapons Delivery		●	
Control Systems		●	
Formation Flight			●
ACM			●
Multi-mission (except HOTAS) (A)			●

(A) Hands on throttle and stick

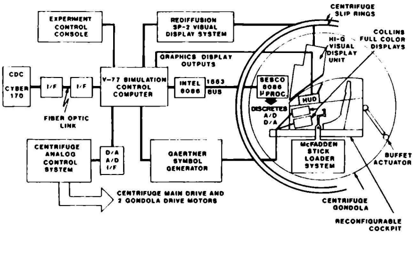

EXPERIMENT CONTROL CONSOLE
REDIFFUSION SP-2 VISUAL DISPLAY SYSTEM
CENTRIFUGE SLIP RINGS
COLLINS FULL COLOR DISPLAYS
HI-G VISUAL DISPLAY UNIT
GRAPHICS DISPLAY OUTPUTS
CDC CYBER 170
I/F
I/F
FIBER OPTIC LINK
V-77 SIMULATION CONTROL COMPUTER
INTEL 8086
1553 BUS
SESCO 8086 μPROC
DISCRETES A/D D/A
HUD
CENTRIFUGE ANALOG CONTROL SYSTEM
D/A A/D I/F
GAERTNER SYMBOL GENERATOR
McFADDEN STICK LOADER SYSTEM
BUFFET ACTUATOR
CENTRIFUGE MAIN DRIVE AND 2 GONDOLA DRIVE MOTORS
CENTRIFUGE GONDOLA
RECONFIGURABLE COCKPIT

FIGURE 1 - DFS BLOCK DIAGRAM

"I feel more comfortable in this (the DFS) than in the (fixed base) domed simulator."

"The departure cues; rolling motion with nose falling through, are excellent."

"Considering the limitations of the concept, the simulator was well designed and executed...Incipient spins through fully developed spins were very well simulated."

"G-feel was good above 2G steady state and onset."

"The simulated - 4Gx spin was qualitatively similar to what I experienced in an actual F-14 flat spin."

The DFS's overall flight fidelity is not as good as that of the 2F95 (F-14 OFT) or the 2E6 (ACM Simulator). However, "as a departure/spin trainer it is far superior."

"The simulator was marginally similar to the F-14A at airspeeds below 250 KIAS. At higher airspeeds, the simulation was not representative. Departure and spin were very realistic."

The DFS provides "extremely realistic simulation of flying...outside references were a very significant factor in the flight simulation. Onset and use of G very, very realistic."

The visual system realism was "one of the most beneficial elements of the trainer. Queasiness (was) overcome by looking "outside" as in a real airplane."

"Positive g's are excellent. (g reduction) comes on faster than a real aircraft, making forward stick sensitive."

FIGURE 2 - PILOT QUALITATIVE COMMENTS

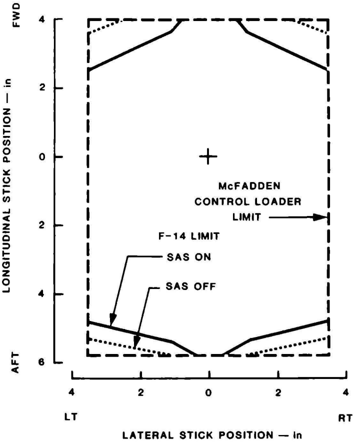


FIGURE 3 - DFS COCKPIT CONTROL POSITION LIMITS

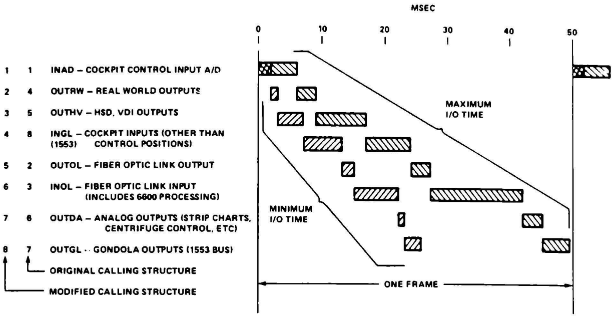


FIGURE 4 - SCC INPUT/OUTPUT HIERARCHY

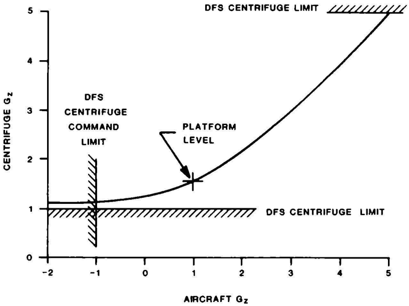


FIGURE 5 - CENTRIFUGE G_Z COMMAND FUNCTION

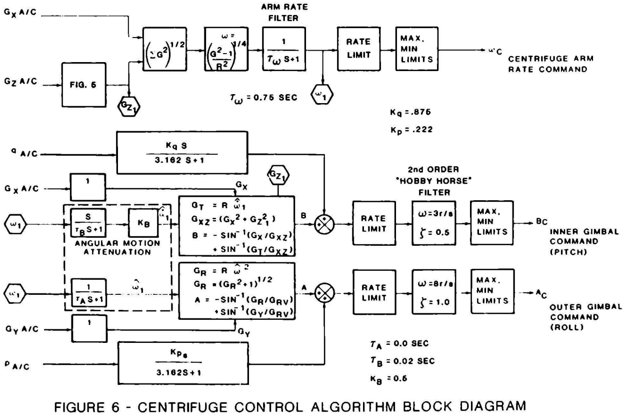

Gx A/C
Gz A/C
FIG. 5
Gz1
$(\Sigma G^2)^{1/2}$
$\omega = \left(\frac{G^2-1}{R^2}\right)^{1/4}$
ARM RATE FILTER
$\frac{1}{T_\omega S+1}$
RATE LIMIT
MAX. MIN LIMITS
ωC
CENTRIFUGE ARM RATE COMMAND
T_ω = 0.75 SEC
ω1
Kq = .876
Kp = .222
q A/C
$\frac{K_q S}{3.162 S+1}$
Gx A/C
1
Gx
Gz1
ω1
$\frac{S}{T_B S+1}$
KB
ANGULAR MOTION ATTENUATION
$G_T = R\,\hat{\omega}_1$
$G_{XZ} = (G_X^2 + G_{Z1}^2)$
$B = -\sin^{-1}(G_X/G_{XZ}) + \sin^{-1}(G_T/G_{XZ})$
B
2nd ORDER "HOBBY HORSE" FILTER
RATE LIMIT
ω=3r/s
ζ = 0.5
MAX. MIN LIMITS
BC
INNER GIMBAL COMMAND (PITCH)
ω1
$\frac{1}{T_A S+1}$
$\hat{\omega}_1$
$G_R = R\,\hat{\omega}^2$
$G_R = (G_R^2+1)^{1/2}$
$A = -\sin^{-1}(G_R/G_{RV}) + \sin^{-1}(G_Y/G_{RV})$
A
RATE LIMIT
ω=8r/s
ζ = 1.0
MAX. MIN LIMITS
AC
OUTER GIMBAL COMMAND (ROLL)
Gy A/C
1
Gy
p A/C
$\frac{K_{P_s}}{3.162S+1}$
T_A = 0.0 SEC
T_B = 0.02 SEC
KB = 0.5

FIGURE 6 - CENTRIFUGE CONTROL ALGORITHM BLOCK DIAGRAM

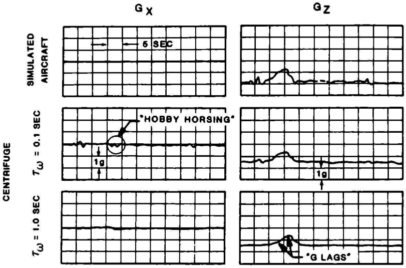


FIGURE 7 - PITCH PULSE ACCELERATION RESPONSES

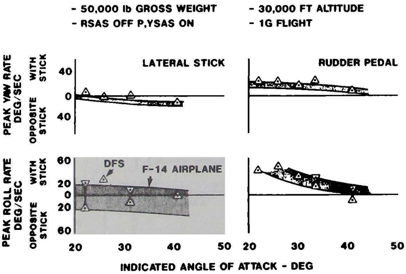


FIGURE 8 - F-14A/DFS HIGH AOA MANEUVERING FIDELITY

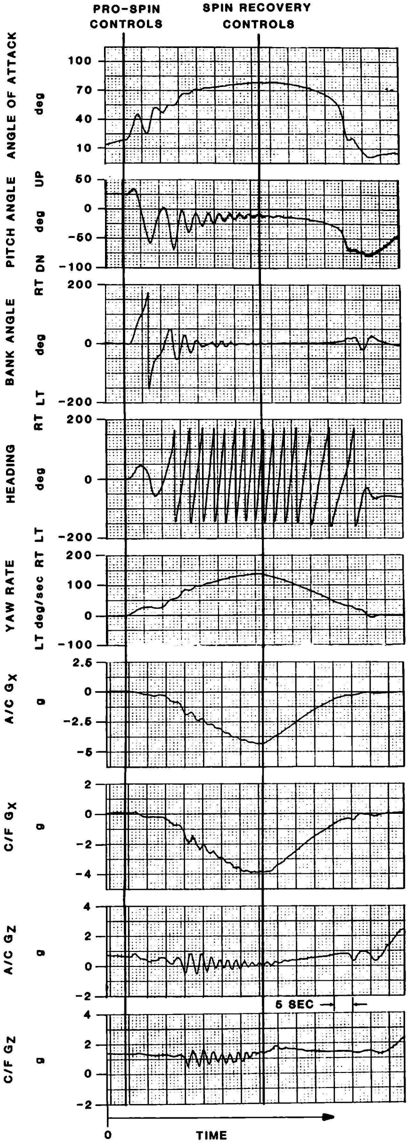


FIGURE 9 - 4 G_X SPIN SIMULATION

85-1742

CONTROLLING THE HUMAN CENTRIFUGE AS A FORCE AND MOTION PLATFORM FOR THE DYNAMIC FLIGHT SIMULATOR

Richard J. Crosbie*
VEDA INCORPORATED
Warminster, PA 18974
and
Dennis A. Kiefer**
NAVAL AIR DEVELOPMENT CENTER
Warminster, PA 18974

Abstract

A major challenge in the development of the Naval Air Development Center's Dynamic Flight Simulator (DFS) was the design of an appropriate control algorithm for the three DOF. Centrifuge. Its design goal was to create a sense of angular motion realism for the DFS pilot while immersing him in the total G force environment of his simulated aircraft. This paper presents the method used in the design and development of the control algorithm, the specific details of its implementation, and DFS flight data which illustrate its performance during various pilot controlled maneuvers. The approach used in the design emphasized perceptual rather than physical realism. Experimentally derived mathematical models of the human proprioceptive system provided invaluable assistance during the development and testing periods. These models enabled the use of linear and angular accleration cues in combination to achieve the design goal of the algorithm. The models also provided assistance during the final tuning of the algorithm with the pilot in the control loop by isolating and monitoring the source of undesirable angular motion stimuli while control parameter adjustments were made. The potential application of the method used here to the design of control algorithms for other motion platforms is discussed.

Introduction

The completion of the Naval Air Development Center's Dynamic Flight Simulator (DFS) has extended the force generation capability of ground based flight simulation beyond that of any previous simulation [1]. Using the Center's unique Centrifuge as a force and motion base, the DFS is capable of generating, under pilot control, the total multi-directional G force environment associated with controlled and uncontrolled flight of current and future high performance aircraft. A major challenge in the development of the DFS was the design of an appropriate centrifuge control algorithm.

A previous paper on this subject, written prior to placing the pilot in the DFS/Centrifuge control loop, presented a preliminary design for the control algorithm and the results of some limited open-loop control tests.[2]. After a year of successful use of the current control algorithm in the operation of the DFS, it can now be generalized to aid in the design of control algorithms for other force/motion platforms.

Nomenclature

Θ_{ac}, Θ_{aa}	DFS, Aircraft pilot perceived angular response in pitch to pure angular rotation
Φ_{ac}, Φ_{aa}	DFS, Aircraft pilot perceived angular response in roll to pure angular rotation
Θ_{vc}, Θ_{va}	DFS, Aircraft pilot perceived angular response in pitch to pure vector rotation
Φ_{vc}, Φ_{va}	DFS, Aircraft pilot perceived angular response in roll to pure vector rotation
Θ_c, Θ_a	DFS, Aircraft pilot perceived angular response in pitch to combined angular and vector rotation
Φ_c, Φ_a	DFS, Aircraft pilot perceived angular response in roll to combined angular and vector rotation
θ_{vc}, θ_{va}	Acceleration vector angular displacement about the DFS, Aircraft pilot's pitch axis
ϕ_{vc}, ϕ_{va}	Acceleration vector angular displacement about the DFS, Aircraft pilot's roll axis
$A,\dot{A}$; $B,\dot{B}$	Gimbal angular displacement, velocity about the DFS pilot's roll and pitch axes
$\Omega_c(p_c,q_c,r_c)$	DFS angular velocity about pilot's roll, pitch and yaw axes
$\Omega_a(p_a,q_a,r_a)$	Aircraft angular velocity about pilot's roll, pitch and yaw axes
R	Radius of Centrifuge arm (50 ft)
ω, $\dot{\omega}$	Angular velocity, angular acceleration of Centrifuge
s	Laplace operator
G_R	Radial acceleration of Centrifuge arm in multiples of g (accel. of gravity), $R\omega^2/g$

*Senior Applications Engineer, Member AIAA
**Project Engineer

G_T	Tangential acceleration of Centrifuge arm, Rω/g
G_v	Vertical acceleration of Centrifuge arm, g/g = 1
G_{xc}, G_{xa}	Transverse acceleration on DFS, Aircraft pilot
G_{yc}, G_{ya}	Lateral acceleration on DFS, Aircraft pilot
G_{zc}, G_{za}	Longitudinal acceleration on DFS, Aircraft pilot
G_c, G_a	Resultant acceleration on DFS, Aircraft pilot
G_c^{1}	G_c delayed by approximately one second
G_{xzc}	$G_{xc}^{2} + G_{zc}^{2}$
G_{yzc}	$G_{yc}^{2} + G_{zc}^{2}$
G_{Rv}	$G_R^{2} + G_v^{2}$

Background

Previous studies have shown that pilot control problems are introduced when the centrifuge control of the DFS is programmed to precisely create the G force environment of high performance aircraft during the transient phase of their G maneuvers. The centrifuge is limited to three controllable degress of freedom (DOF); the two angles of the dual gimbal system in which the DFS cockpit is mounted and the velocity of the centrifuge arm to which the gimbal system is attached. Although these three DOF are essentially unlimited in their range of travel and the three drive motors are highly responsive, the DFS is confined to move in a horizontal plane about a fixed radius circle. This confinement dictates the physical impossibility for the DFS to match both the linear and angular accelerations of an aircraft which flies in an unlimited six DOF flight regime.

This limitation has not been a problem during the many years in which the centrifuge has successfully been used for physiological research studies. Here the requirement was to precisely generate, under open-loop control, a wide variety of complex multi-directional G profiles on a subject, including those having rapid G onset rates up to 10 G/sec. The limitation does become a problem, however, when the centrifuge is used as a force and motion base for the DFS and the subject is a pilot placed in the control loop. Rapid angular motions of the two gimbals and the centrifuge arm required to precisely generate rapidly changing G profiles are totally unrelated to those of the simulated aircraft. This difference would create a conflict in the pilot's sensory response to his control inputs which would make the DFS difficult to fly and would severely limit its use for pilot controlled flight simulation studies.

Precise G force control of the DFS is also difficult in the flight regime slightly above the +1 G level because of the large gimbal motions required there and impossible between +1 G and -1 G because of the ever present acceleration of gravity. Therefore, in order that the DFS be flyable and pilot acceptable over the full range of its capability, the requirement for precise G force control must be relaxed during periods of rapidly changing G and in the region slightly above and below 1 G. This relaxation, however, must not compromise the basic purpose of the DFS, i.e., to generate a force field which would have the same debilitating and disorientating effect on the DFS pilot as would the force field of his simulated aircraft. Moreover, the DFS pilot must be provided with angular motion cues which do not conflict in phase or amplitude with those of his visual field.

The problem is illustrated by referring to Figure 1. Here the visual and motion closed-loop human sensing system is depicted as it affects both the aircraft pilot and the DFS pilot in controlling their respective vehicles. It is assumed that the DFS flight characteristics, as perceived by the DFS pilot through the forces on his control stick and the response of his instruments and visual display system, are accurately representative of the aircraft. The problem remaining, therefore, is to create a linear and angular force and motion environment which the DFS pilot would similarly perceive as representative of the aircraft. This requires that signals representative of the aircraft linear forces G_a, and angular rates Ω_a generated in real time by the DFS computer, be used to drive the centrifuge control signals through an appropriate control algorithm.

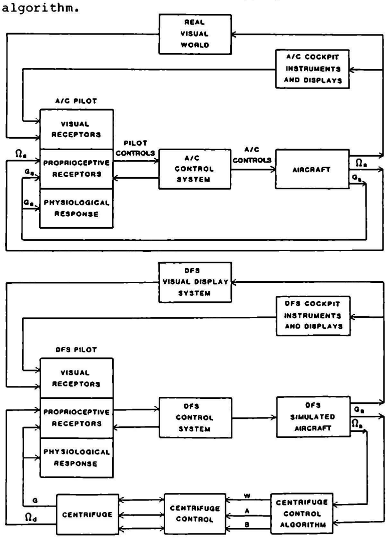


FIGURE 1 - VISUAL AND MOTION CLOSED-LOOP HUMAN SENSING SYSTEM FOR AIRCRAFT PILOT AND DFS PILOT.

Approach

As has been previous stated, it is physically impossible to simultaneously generate linear forces and angular motions on the centrifuge which match those of the aircraft. Also, it is undesirable to match either the linear forces or the angular motions independent of the other. The design of the control algorithm, therefore, must compromise both requirements in order to achieve the optimum effect. Of prime concern in this compromise, is to effect a match between the angular motion information perceived by the DFS pilot through his visual receptors and that perceived through his proprioceptive receptors. The proprioceptive receptors include both the tactile/kinesthetic receptors and the vestibular receptors. Effecting such a match would also require that a match exist between the angular motions perceived by the DFS pilot and the aircraft pilot to the same control inputs.

Fortunately, this match can be effected without requiring that the angular motions of the DFS match those of the aircraft. Human perception of roll and pitch angular motion has been shown to be influenced by the combined action of angular rotation and rotating linear acceleration vectors about the roll and pitch axes, respectively [2]. Consequently, by controlling the roll and pitch gimbal drive signals to create rotating linear acceleration vectors in a timely fashion with respect to the pure angular rotations, the desired angular motion perception can be achieved.

In descending order of their development, the design goals of the centrifuge control algorithm were:

1) To provide a control logic which permits smooth and continuous control of the DFS in the region of 1 G and below.

2) To minimize the disturbing angular artifacts experienced by DFS pilots during periods of varying G while generating a G environment which produces the desired stress effect on his physiological system.

3) To effect a match between the angular motion information perceived by the DFS pilot through his visual receptors and that perceived through his proprioceptive receptors.

4) To limit operation of the DFS below safe structural and physiological limits.

5) To filter high frequency drive signals to the gimbal drive motors with a minimum phase shift in order to prevent shocking the gimbal drive linkages into oscillation.

The initial form of the centrifuge control algorithm involved the mechanization of the centrifuge and gimbal drive control signals used to achieve precise G force control. Note that acceleration washout circuitry, normally used in the control of motion platforms with limited excursions, is not necessary in the control of the centrifuge. These drive signals were then modified to emphasize perceptual rather than physical realism. Experimentally derived mathematical models of the human proprioceptive system were used throughout the development to define the shape and phase of the control perturbations and to adjust and fine-tune the controls for maximum pilot acceptance. These models defined human perceptual angular response to the individual and combined stimuli of angular and linear accelerations.

Description

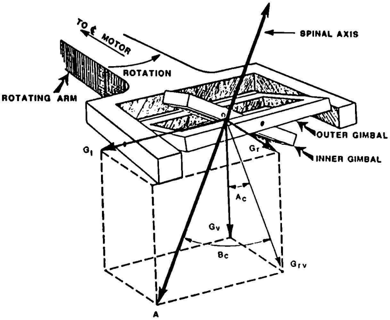


FIGURE 2 - CENTRIFUGE GIMBAL CONSTRUCTION SHOWING GIMBAL DISPLACEMENT ANGLES FOR COORDINATED G OPERATION.

The NAVAIRDEVCEN centrifuge has an enclosed 10 feet spherical gondola mounted in a controllable two gimbal system at the end of a 50 ft long main arm. The gimbal system (Figure 2) consists of an outer gimbal which rotates through an angle A about a horizontal axis perpendicular to the centrifuge arm, and an inner gimbal which rotates through an angle B about an axis in the plane of the outer gimbal and perpendicular to its axis. These two quantities, A and B, along with ω, the angular velocity of the centrifuge arm, define the three independent controllable functions which are used in the programming and control of the Centrifuge. They are related to the three component linear accelerations G_{xc}, G_{yc} and G_{zc} and the three component angular velocities p_c, q_c and r_c experienced by a centrifuge subject through the following equations:

1, $G_{xc} = G_T \text{ Cos } B + G_R \text{ Sin } A \text{ Sin } B - G_V \text{ Cos } A \text{ Sin } B$

2, $G_{yc} = G_R \text{ Cos } A + G_V \text{ Sin } B$

3, $G_{zc} = G_T \text{ Sin } B - G_R \text{ Sin } A \text{ Cos } B + G_V \text{ Cos } A \text{ Cos } B$

4, $p_c = \dot{A} \text{ Cos } B - \omega \text{ Cos } A \text{ Sin } B$

5, $q_c = \dot{B} + \omega \text{ Sin } A$

6, $r_c = \dot{A} \text{ Sin } B + \omega \text{ Cos } A \text{ Cos } B$

Solving equations 1, 2, and 3 yields the following control equations:

$$7,\ \omega = \frac{g}{R}\ {}^{4}\ (G_c^{\,2} - 1) - G_T^{\,2} \sim .802\ {}^{4}\ (G_c^{\,2} - 1)^{*}$$

*Approximate solution - selected for increased G response.

$$8,\ A = -\mathrm{Sin}^{-1}\ \left(\frac{G_R}{G_{Rv}}\right) + \mathrm{Sin}^{-1}\ \left(\frac{G_{yc}}{G_{Rv}}\right) = A_c + \Delta A_y$$

$$9,\ B = \mathrm{Sin}^{-1}\ \left(\frac{G_T}{G_{xzc}}\right) - \mathrm{Sin}^{-1}\ \left(\frac{G_{xc}}{G_{xzc}}\right) = B_c + \Delta B_x$$

where A_c and B_c are gimbal roll and pitch angles required to coordinate the subject with respect to the resultant G_c vector, i.e., $G_{xc} = G_{yc} = 0$. The ΔA_y and ΔB_x quantities are additions to these angles required to produce component G_{yc} and G_{xc} accelerations on the subject respectively. The relationship between the gimbal angles and the subject component accelerations are shown in Figures 3 and 4. The A_c and B_c angular motions are purely centrifuge related and are considered to be the prime contributors to the previously mentioned angular artifacts perceived by centrifuge subjects. A major design goal of the control algorithm, therefore, was to alter these drive signals to minimize their effect on creating these artifacts.

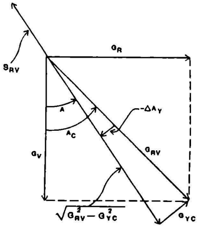


FIGURE 3
CENTRIFUGE ACCELERATIONS IN CENTRIFUGE RV PLANE
S_{rv} IS COMPONENT OF SUBJECT'S SPINAL AXIS IN RV PLANE.

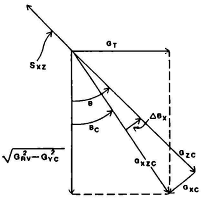


FIGURE 4
CENTRIFUGE ACCELERATIONS IN SUBJECT'S XZ PLANE.
S_{xz} IS COMPONENT OF SUBJECT'S SPINAL AXIS IN XZ PLANE.

Control Algorithm Development:

The scheme used to develop the centrifuge control algorithm for DFS application was to modify or perturb the basic force equations 7, 8, and 9 in order to achieve a pilot acceptable force and motion environment. An analysis of these equations reveals that they provide no solution for the region between ±1 G and they develop large gimbal motion commands, A and B, in the region slightly above 1 G. Since either of these features would preclude pilot controlled operation of the DFS/centrifuge during increasing or decreasing G maneuvers from straight and level flight, some immediate changes were mandatory.

The concept of operating the DFS at a bias G level provided the obvious solution. However, the exact amount of the bias and the slope of the G curve above and below the bias level had to be determined. The curve shown in Figure 5 with a 1.55 G_z bias level was finally selected as it satisfied the following criteria:

1) Pilot acceptable G_{zc} level with regard to fatigue for continuous operation. (Typically 1 hour.)

2) Minimal gimbal motions during G increases or decreases from the bias level.

3) Continuous control of the DFS to -1 G_{za} with some G unloading effects and with no disturbing discontinuities.

4) Continuous turning of centrifuge during DFS operation (-1 G_{za} = 1.1 G_{zc}).

5) Primarily effects the G_{zc}. G_{xc} and G_{yc} are minimally effected.

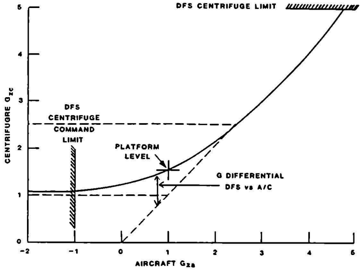


FIGURE 5 - CENTRIFUGE G_z COMMAND FUNCTION

In designing the control algorithm, knowledge of the pilot's perceptual angular response to the individual and combined stimuli of rotating linear acceleration vectors and angular accelerations proved invaluable. The following equations, experimentally derived on the centrifuge and reported in [2], define human perceptual response to the individual component stimuli about the pitch and roll axes:

$$10,\ \Phi_{ac} = \frac{A}{(G_c^{\,1} + 1)}\ \frac{12.48\ S}{(S + 10)(S + 5)}$$

$$= \frac{A}{(G_c^{\,1} + 1)}\ T_{\phi a}$$

$$11,\ \Theta_{ac} = \frac{B}{(G_c^{\,1} + 1)}\ \frac{12.48\ S}{(S + 10)(S + 2.5)}$$

$$= \frac{B}{(G_c^{\,1} + 1)}\ T_{\theta a}$$

$$12,\ \Phi_{vc} = \frac{G_c^{\,1}\ \phi_{vc}}{(G_c^{\,1} + 1)}\ \frac{6.66\ (S + .75)}{(S + 10)(S + .5)}$$

$$= \frac{G_c^{\,1}\ \phi_{vc}}{(G_c^{\,1} + 1)}\ T_{\phi v}$$

$$13,\ \Theta_{vc} = \frac{G_c^{\,1}\theta_{vc}}{(G_c^{\,1} + 1)}\ \frac{2.66\ (S + .75)}{(S + 10)(S + .2)}$$

$$= \frac{G_c^{\,1}\theta_{vc}}{(G_c^{\,1} + 1)}\ T_{\theta v}$$

In [2] it was further reported that human perceptual angular response to the combined stimuli of rotating linear acceleration vectors and angular accelerations is provided by summing the response to the individual stimuli as shown by the following equations:

$$14,\ \Phi_c = \Phi_{ac} + \Phi_{vc} = \frac{A\ T_{\phi a} + G_c^{\,1}\Phi_{vc}T_{\phi v}}{G_c^{\,1} + 1}$$

$$15,\ \Theta_c = \Theta_{ac} + \Theta_{vc} = \frac{B\ T_{\theta a} + G_c^{\,1}\theta_{vc}T_{\theta v}}{G_c^{\,1} + 1}$$

The implications of equations 14 and 15 are that the effects of combining the stimuli are not only additive and reinforcing as they are in a normal environment, but that they are also subtractive. That is, one component angular stimulus can be used to counter influence the pilot's perception of the other.

The application of equations 10 - 15, as they were used in the development of the centrifuge control algorithm, is graphically illustrated in Figures 6 - 9. These DFS flight data recordings provide comparisons between the roll angular responses predicted for the DFS pilot during a series of roll left and roll right maneuvers with those predicted for the aircraft pilot during the same maneuvers. (A similar set of control and response data were obtained for the pitch axis, but will not be shown here for the sake of brevity.) In order to minimize the number of dynamic manned runs required for the algorithm development, the flight data were initially recorded with the DFS in a static mode, i.e., the centrifuge not moving. The recordings were then played back through the centrifuge control algorithm to drive the unmanned DFS/centrifuge in a dynamic mode and provide the data shown in Figures 6 - 9. Figure 6 represents the results of using the "force control" method for driving the centrifuge. This method replaces all DFS accelerations with aircraft accelerations in equations 7, 8, and 9 except for the G_z command which is further

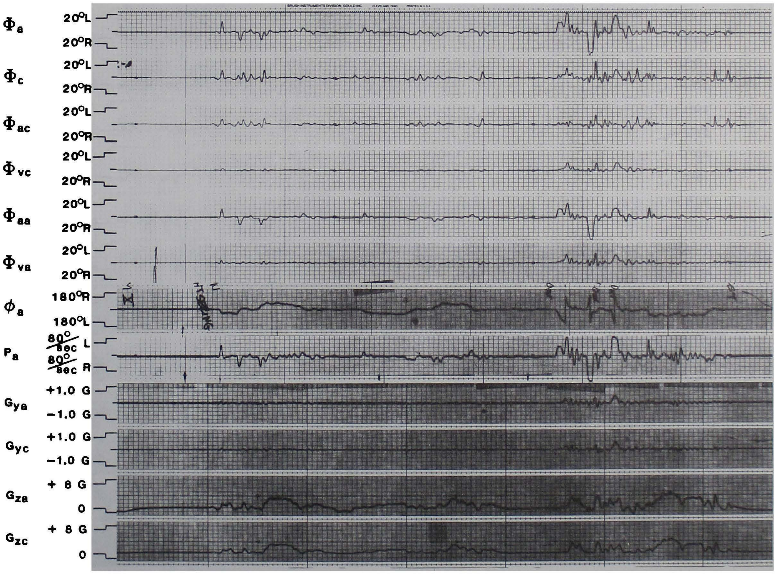


FIGURE 6 - DFS FLIGHT DATA RECORDINGS USING PRECISE 6 FORCE CONTROL. NOTE THE EXCELLENT MATCH BETWEEN THE DFS AND AIRCRAFT ACCELERATIONS AND THE MISMATCH BETWEEN THE PERCEIVED ANGULAR RESPONSES, Φ_c and Φ_a, PREDICTED FOR THE TWO PILOTS.

modified as shown in Figure 5, and computes A, B, and ω. Here the measured G_{zc} and G_{yc} accelerations of the DFS are seen to match quite well with those of the aircraft, G_{za} and G_{ya}, as does also the predicted vector response data of the DFS pilot, Φ_{vc}, with that of the aircraft pilot Φ_{va}. However, the predicted angular response data of the DFS pilot, Φ_{ac}, does not match that of the aircraft pilot, Φ_{aa}, resulting in a mismatch between the resultant roll response data of the two pilots, Φ_c and Φ_a, and by inference between the DFS pilot's visual and proprioceptive receptors. This data is consistent with the previously mentioned angular artifacts produced during precise G force control of the DFS. Noteworthy in the acceleration data is the smooth and continuous performance of the DFS as the aircraft dips below 1.0 G_{za}.

In order to reduce these angular artifacts, the initial strategy was to minimize the pilot's perception of pitch and roll angular motions during the transient periods of coordinated G_{zc} profiles, i.e., $G_{xc} = G_{yc} = 0$. (Again for the sake of brevity, the following derivation will refer only to angular motions about the pilot's roll axis.) This requires that a roll gimbal command function, A_o, be derived which will minimize Φ_c in equation 14. Let A_o be derived by perturbing A_c, the gimbal command function described in equation 8 and used for precise G force control when $G_{yc} = 0$.

Then,

16, $A_o = A_c + \Delta A_o = A_c + \phi_{vc_o}$

where ΔA_o is the vector command function $\phi_{vc,o}$ created by perturbing A_c. Substituting these quantities into equation 14 and equating to zero provides:

17, $A_o\, T_{\phi a} + G_c^{\,1}\, \phi_{vc_o}\, T_{\phi v} = 0$

Solving equations 16 and 17 yields

18, $$A_o = \frac{A_c}{1 + \dfrac{T_{\phi a}}{G_c^{\,1}\, T_{\phi v}}} \sim \frac{A_c}{1 + .25\ S}$$

It should be noted that this 0.25 sec delay time on A_c is in addition to the 0.4 sec delay time of the centrifuge G response. Taking into account also the 0.2 sec delay time of the gimbal itself, the actual delay time for the A_c is 0.45 sec.

With the aircraft G_{ya} removed from the G_c profile, the above approximate solution for A_o was used to drive the roll gimbal and minimize ϕ_c. The minimizing effect, as shown in Figure 7, was

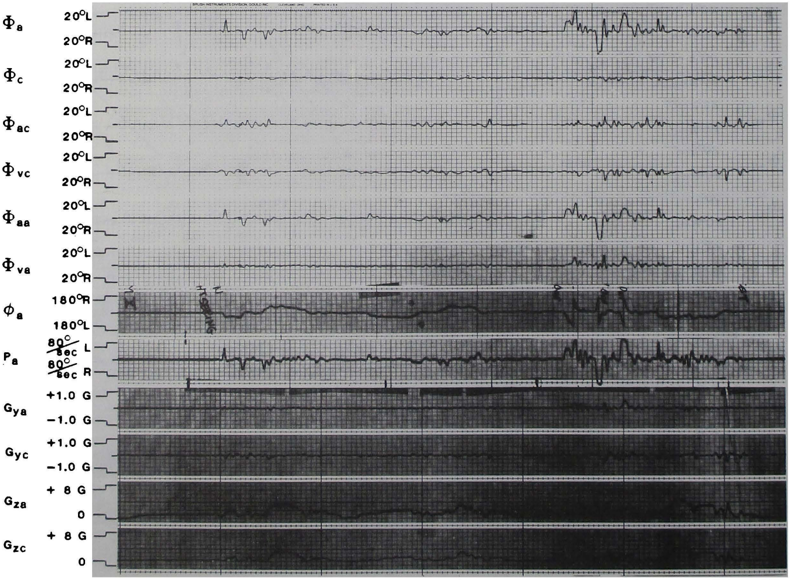


FIGURE 7 - DFS FLIGHT DATA RECORDINGS WITH G_{ya} REMOVED AND CENTRIFUGE ROLL GIMBAL CONTROL ALTERED TO MINIMIZE Φ_c. NOTE THE CANCELLING EFFECT BETWEEN Φ_{ac} AND Φ_{vc}.

essentially achieved. Note the cancelling effect which occurs between the DFS pilot's Φ_{ac} and Φ_{vc} responses.

The next step in the algorithm development was to insert the aircraft G_{ya} accelerations into the A drive signal. This is the ΔA_y of equation 8. The effect of this insertion, as shown in Figure 8, adds the same angular and vector motion cues to the DFS pilot as to the aircraft pilot.

Finally, in order to introduce aircraft angular motion cueing to the DFS pilot, the centrifuge's roll gimbal control signal must be further altered to include a function of the aircraft's angular roll motion. In order that the DFS pilot perceive the same angular roll motion cues as the aircraft pilot following this alteration, the following equation must be satisfied:

19, $\Phi_{aa} + \Phi_{va} = \Phi_{ac} + \Phi_{vc}$

or

20, $(\frac{P_a}{S})\ T_{\phi a} + G_c^{\ 1}\ \phi_{va}\ T_{\phi v} = A\ T_{\phi a} + G_c^{\ 1}\ \phi_{vc}\ T_{\phi v}$

where $^{P}a/S$ is the roll angular displacement of the aircraft.

Let:

21, $A = A_o + \Delta A_y + \Delta A_a$

where ΔA_a is the aircraft function intended to achieve the proper angular motion effect.

Now recalling from equation 17 that

$A_o\ T_{\phi a} = G_c^{\ 1}\ \phi_{vc_o}\ T_{\phi v}$

and that ΔA_y generates a vector angle which has the same perceived effect on the DFS pilot as on the aircraft pilot, the substitution of equation 21 into equation 20 provides:

22, $(\frac{P_a}{S})\ T_{\phi a} = \Delta A_a\ T_{\phi a} + G_c^{\ 1} (\phi_{vc} - \phi_{vc_o})\ T_{\phi v}$

Now since by definition,

$\Delta A_a = (\phi_{vc} - \phi_{vc_o})$

23, $\Delta A_a = \dfrac{(\frac{P_a}{S})\ T_{\phi a}}{T_{\phi a} + G_c^{\ 1} T_{\phi v}} = \dfrac{P_a}{1 + \dfrac{T_{\phi a}}{G_c^{\ 1}\ T_{\phi v}}} \sim \dfrac{P_a}{1 + .25\ S}$

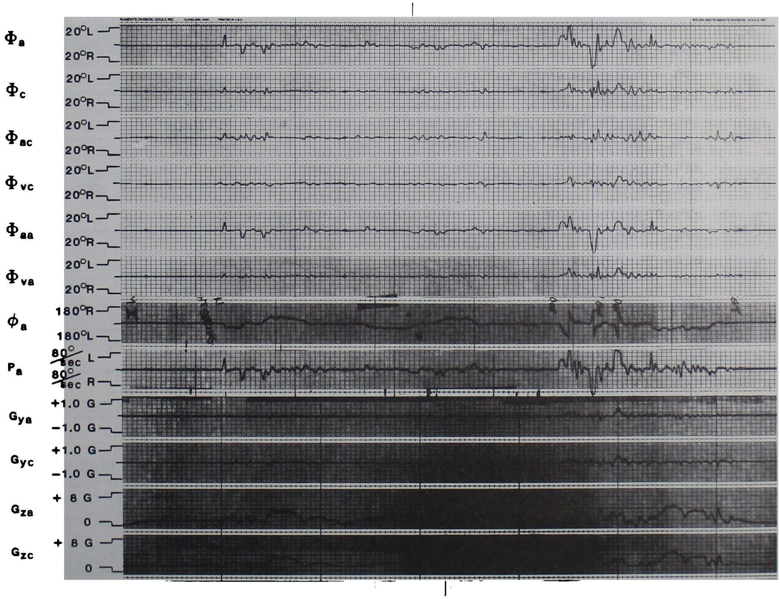


FIGURE 8 - DFS FLIGHT DATA RECORDINGS SIMILAR TO FIGURE 7 WITH G_{ya} INSERTED INTO THE CONTROL SIGNAL.

Thus finally, the total roll gimbal drive signal is defined as:

$$24, \quad A = \frac{A_c}{1 + .25\ S} + \Delta\ A_y + \frac{P_a}{1 + .25\ S}$$

where $A_c = -\mathrm{Sin}^{-1}(\frac{G_R}{G_{Rv}})$; and $\Delta A_y = \mathrm{Sin}^{-1}(\frac{G_{ya}}{G_{Rv}})$

from equation 8.

Noting that two delay terms on A_c and P_a are not referenced to the same time frame, this equation must be modified to include the response times of the gimbal and arm drive systems. The time delay terms on A_c is referenced to the gimbal response angle required for coordinated G control and must be modified as stated previously. The delay term on P_a is referenced to the aircraft angular velocity and may be ignored to allow for the response time of the gimbal drive system. Thus by driving the centrifuge roll angle, A, with P_a, which is a lead function of the aircraft roll angle, an in-phase response of Φ_c with Φ_a can be achieved in spite of the 0.2 sec delay time of the roll gimbal response. P_a did have to be altered, however, to remove any low frequency contents which could create sustained G_{yc} accelerations. This was done with a high pass filter which had little effect on its phase response. The final form of equation 25 as applied to the recorded DFS flight maneuvers is:

$$25, \quad A = \frac{A_c}{1 + .45S} + \Delta\ A_y + \frac{.22\ P_a S}{1 + 3.16S}$$

The results of applying this equation is shown in Figure 9. Note, the proper phasing and amplitude of the DFS pilot's perceived response in roll compared to the aircraft pilot's, during both roll left and roll right maneuvers and including the "roll over" maneuvers. The price to attain this matching, which is critical for closed-loop pilot control of the DFS, is to sacrifice the matching of the G_y accelerations. These perturbations to the roll gimbal command signal are seen to have little effect on the centrifuge G_{zc} which lags the aircraft G_{za} by approximately 0.4 sec. This lag should not prevent the desired physiological response from occurring in the DFS pilot.

Some additional filtering in the gimbal command signals were required, particularly in the pitch gimbal, to prevent high frequency shocking of the gimbal drive linkage. The large inertial load of the DFS, approximately 2500 pounds within the centrifuge gondola, places a large strain on the linkage, especially during command reversals

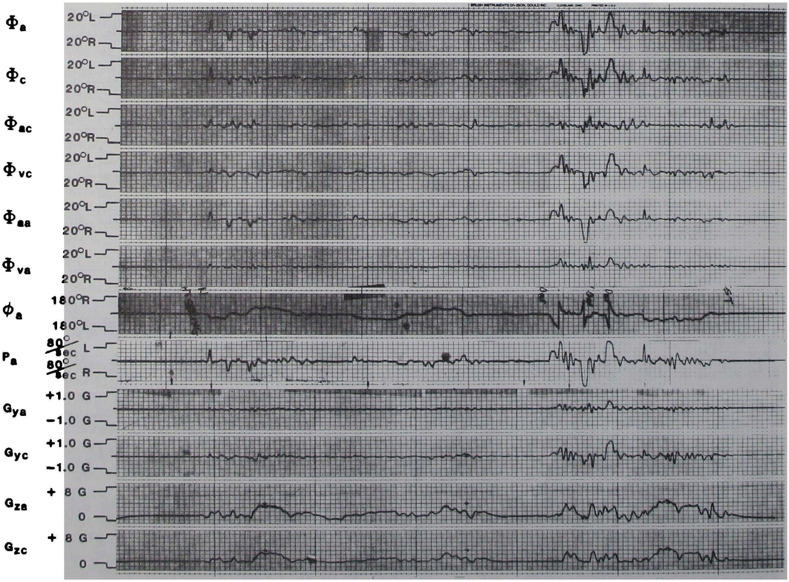


FIGURE 9 - DFS FLIGHT DATA RECORDINGS USING THE CENTRIFUGE CONTROL ALGORITHM IN ITS FINAL OPERATIONAL FORM. NOTE THE EXCELLENT MATCH IN PHASE AND AMPLITUDE BETWEEN Φ_c AND Φ_a AND THE ACCEPTABLE MATCH BETWEEN G_{zc} AND G_{za}. WHAT HAS BEEN SACRIFICED IS A MATCH BETWEEN G_{yc} AND G_{ya} DURING PERIODS OF VARYING G.

under G. Rate limiters are strategically placed on all drive signals to prevent damage to the linkage and G limiters exist in all axes to protect the pilot. An overall block diagram of the current status of the control algorithm is shown in Figure 10.

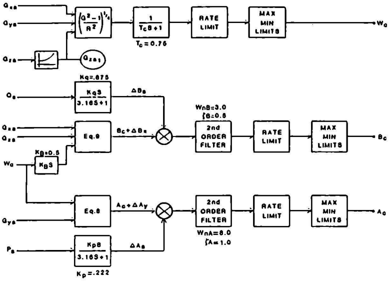


FIGURE 10 - BLOCK DIAGRAM OF CENTRIFUGE CONTROL ALGORITHM.

Pilot comments on the DFS have been uncharacteristically enthusiastic. They enjoy pulling G and have experienced very little motion sickness after flying it for an hour or more, especially after the first session. One pilot noted that it was the first flight simulator he had flown in which the visual and vestibular cues were in agreement. Another pilot commented after $1^1/_2$ hours flying the DFS, that he could not tolerate fixed base simulators with visual systems for more than 15 minutes at a time.

The problems involved in applying this method to the development of control algorithms for other motion systems are far simpler than those encountered here. Angular motions inherent in the operation of the centrifuge are normally not encountered in the operation of other devices. Component linear acceleration and angular motions can be measured during the operation of the simulator and used to drive the human response transfer functions to obtain the predicted angular motions in roll and pitch perceived by the simulator pilot. These can then be compared with those predicted for the aircraft pilot and the control drive signals altered to achieve the best match. Recorded time histories of simulator flight data to provide repeatable control inputs as used here are recommended.

References

1. Crosbie, R. J., Eyth, J., "A Total G Force Environment Dynamic Flight Simulator", Proceedings of the AIAA Aerospace Sciences Meeting, Flight Simulation Session. 10-13 January 1983.

2. Crosbie, R. J., "Application of Experimentally Derived Pilot Perceptual Angular Response Transfer Function", Proceedings of the AIAA Flight Simulators Technologies Conference and Technical Display. 13-15 June 1983.

85-1743

USING HUMAN MOTION PERCEPTION MODELS TO OPTIMIZE FLIGHT SIMULATOR MOTION ALGORITHMS

Karl S. Forsstrom
and
Jennifer Doty*
Northrop Corporation, Advanced Systems Division
Pico Rivera, California

Frank M. Cardullo**
State University of New York
Binghampton, New York

Abstract

As part of a research and development project at Northrop Corporation the authors have developed a simulator motion analysis tool which utilizes human motion perception models to objectively judge the quality of simulator platform motion.

Introduction

Humans perceive motion through the visual system, the vestibular system and proprioceptive systems. These motion sensors synergistically combine to produce the overall impression of "motion". This paper deals with the vestibular motion sensors and the techniques which can be used to improve the performance of simulator platform motion systems. It does not investigate the complex interactions among the visual, vestibular or proprioceptive sensors.

There are many indications from flying qualities design studies that it is important for the simulator pilot to feel or be presented with non-visual motion cues in the simulator. Except for very gentle flight maneuvers, the aircraft's motion and the simulator's motion can never be identical. Flight simulator motion is limited by hardware travel constraints. To overcome these constraints, motion washout logic is used to implement restricted motion commands.

In the past, simulation engineers have tried to provide acceptable motion sensations by comparing and minimizing any differences between the simulator cockpit accelerations and the simulated aircraft accelerations. A second technique was to modify the motion drive parameters based on pilot subjective opinions on the quality of simulator motion. A standardized objective method for optimizing the pilot's perceived motion would be very useful for the refinement of aircraft simulator motion drive algorithms.

This paper presents a methodology which utilizes human motion perception models to analyze and optimize motion algorithms. A set of software tools has been developed which can be used to tune the motion drive algorithms to produce motion commands that induce the sense of realistic aircraft motion in the pilot's physiological motion sensors.

Motion Sensor Physiological Background

The vestibular system of the inner ear is located in the bony labyrinthine structure behind the auditory portion of the ear which is just behind each eye. It is composed of the otolith organs, known as the utricle and saccule, which respond to linear accelerations, and the semicircular canals which respond to angular accelerations. The vestibular system is the human motion sensor system that has been studied and understood the most thoroughly and therefore can be modeled the most accurately of all the motion sensors (see Ref. 10 and 11).

The utricle is oriented horizontally on a plane that is tilted approximately twenty-five degrees upward from the head horizontal reference line. The head horizontal reference line is defined as the imaginary line drawn between the outer corner of the eye and the ear opening. The saccule is oriented along the head vertical reference line. The otoliths act as linear accelerometers that are stimulated by linear accelerations and attitude tilt changes with respect to the gravity vector. The otolithic membrane is imbedded with calcite crystals which make it more dense than the endolymph fluid that surrounds it. When the head experiences a linear acceleration the membrane lags behind due to inertia. When the head is tilted the membrane slides downhill. Any relative motion between the dense membrane and the surrounding fluid triggers the hair cell sensors to send motion signals to the brain. These motion signals are called afferent firing impulses.

There are three semicircular canals, the superior, the posterior and the horizontal. These fluid filled rings are oriented at approximate right angles with respect to each other, with the horizontal canal inclined approximately twenty-five degrees above the head horizontal reference line. The semicircular canal mechanics cause them to act like angular accelerometers at low frequencies and like angular velocity transducers at middle range frequencies. When the head experiences an angular acceleration the endolymph fluid impacts the cupula, a gelatinous wedge that seals the enlarged region of each canal, causing it to move and bend, deflecting the hair cells whose attached nerve fibers send afferent firing impulses to the brain.

Motion Perception Modeling

Motion perception models are programmed in manners very similar to any other physical or biological systems using standard control theory and numerical analysis. Empirically measured data

*Simulation Engineer, Member AIAA

**Assistant Professor, Member AIAA

of the neuro-physiological responses of individual motion sensing organs to motion stimulus is obtained by stimulating organs removed from test animals such as monkeys, frogs and mice. This time domain information, together with data from experiments using human subjects on tilt tables, centrifuges, etc., is analyzed and reduced to frequency domain Laplace transform formats.

Once the motion perception models are in the form of Laplace transform transfer functions, they can easily be integrated into a motion analysis program. In addition to providing greater insight into the effectiveness of the motion cueing algorithms, these perception models offer the possibilities of optimizing the cueing algorithms. Through the use of modern optimal control theory, the motion cueing algorithms can be tuned to produce minimum perceived motion errors. Since there are numerous human motion sensors, a Kalman filter technique might be required to extract a composite perceived motion signal which could be used to continuously and optimally vary the internal gains and washout frequencies of the motion cueing algorithms.

Much experimental research in human motion perception and most of the associated literature has been done at the Massachusetts Institute of Technology Man-Vehicle Laboratory in Cambridge, Massachusetts. Young, Ormsby, Oman and Borah form a partial list of the researchers that have been involved with motion perception studies at the MIT lab. The motion perception models discussed in this paper are based on those of Joshua Borah.

The otoliths sense specific force as do linear accelerometers. Borah modeled the otoliths as mechanical accelerometers with the addition of a perception threshold along with some added rate sensitivity to account for afferent processing. The two sets of otolith organs have been replaced by a single cyclopian set located at the center of the head for modeling purposes. The otolith model used is shown in Figure 1a. This model is the same as that used in Borah's 1977 report, "Sensory Mechanism Modeling" (see Ref. 1). The time response of this model to a 0.2 G step input is shown in Figure 1b. The output of the model is an incremental change in the afferent firing rate (impulses per second) from a non-accelerating steady state afferent firing rate. The effect of the perception threshold can be seen by comparing Figure 1b (threshold included) with Figure 1c (threshold excluded).

The semicircular canals are modeled as a highly overdamped torsion pendulum with some added rate sensitivity, adaptation parameters and a perception threshold. A single cyclopian system located at the center of the head has again been used here for modeling purposes. The transfer function used in the semicircular canal model differs from Borah's only in the overall gain value. We use a gain of 0.01 obtained through re-derivation of the transfer function from Ormsby (see Ref. 9). Borah uses a gain of 0.574. The semicircular canal model and its time response to a 1.0 degree/second/second step input are shown in Figures 2a and 2b. Again, the output is afferent firing rate in impulses per second.

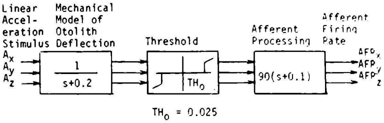


Figure 1a Otolith Model

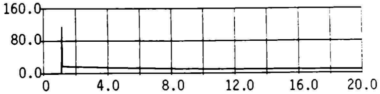


Figure 1b Otolith Model Response (impulses/sec) to a 0.2 G Step

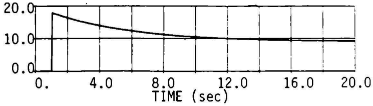


Figure 1c Otolith Model Response (impulses/sec) to a 0.2 G Step (no threshold)

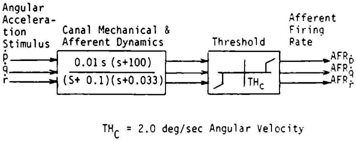

Angular Accelera- tion Stimulus
Canal Mechanical & Afferent Dynamics
Threshold
Afferent Firing Rate
$\dot{p}$
$\dot{q}$
$\dot{r}$
$\frac{0.01 s (s+100)}{(S+ 0.1)(s+0.033)}$
TH_C
$AFR_{\dot{p}}$
$AFR_{\dot{q}}$
$AFR_{\dot{r}}$
$TH_C = 2.0$ deg/sec Angular Velocity

Figure 2a Semicircular Canal Model

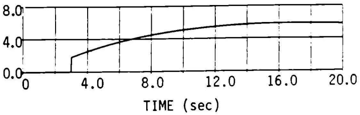


Figure 2b Semicircular Canal Model Response (impulses/sec) to a 1.0 deg/sec^2 Step

Tested and validated motion perception models provide an unbiased measure of perceived motion. Traditionally, the human pilot subject is not a reliable and consistent source of quantitative or even qualitative data. Additionally, perception models allow off-line analysis of motion cueing algorithms without tying up expensive simulator and real-time computer resources.

Comparison of the motion cueing algorithms using the human perception models brings out the positive and negative attributes of each approach. For example, the linear models are generally good for small motion inputs but will create anomalies with large inputs because of linear or angular motion excursion limitations. A nonlinear adaptive model is capable of producing larger amplitude cues because it makes better use of the available platform travel limits, but may produce inconsistent cues since the response is dependent on the initial platform position. The relative phasing of the different motion cueing algorithms becomes readily apparent, especially when one compares the perceived motion signals. The human variability factor can be removed from the motion analysis problem through the use of realistic human motion perception models.

Motion Analytical Tool

The motion analysis program is a tool developed to facilitate simulator motion algorithm optimization. Figure 3 is a functional block diagram of the program. The software consists of a collection of Fortran subroutines that are grouped together to form five modular subsystems:

1) aircraft parameter input module
2) candidate motion cue shaping algorithms and simulator motion commands
3) simulator platform hardware dynamics module
4) human motion perception models
5) a simple output time history plotting routine

The first subsystem, the aircraft parameter input module, is divided into two sections: the motion analysis input driver and the linear aircraft mathematical model. The motion analysis input driver enables the motion cue shaping algorithm or the motion perception models to be driven by a direct input consisting of a step, ramp or sine wave input signal. The linear aircraft mathematical model computes the attitudes, forces and moments of a simplified commercial transport type aircraft. Either the motion input driver or the aircraft math model, whichever the user selects, provides aircraft parameter input values to the candidate motion cue shaping algorithm. These parameters can also bypass the motion algorithm and be routed directly to the human motion perception models.

The second subsystem consists of candidate motion cue shaping algorithms and simulator motion commands. The motion cue shaping algorithm uses motion washout logic to compute simulator motion commands whose limits are defined by the simulator motion hardware. The candidate motion algorithms being investigated are a conventional linear washout filter, a non-linear adaptive washout

MOTION ANALYSIS PROGRAM BLOCK DIAGRAM

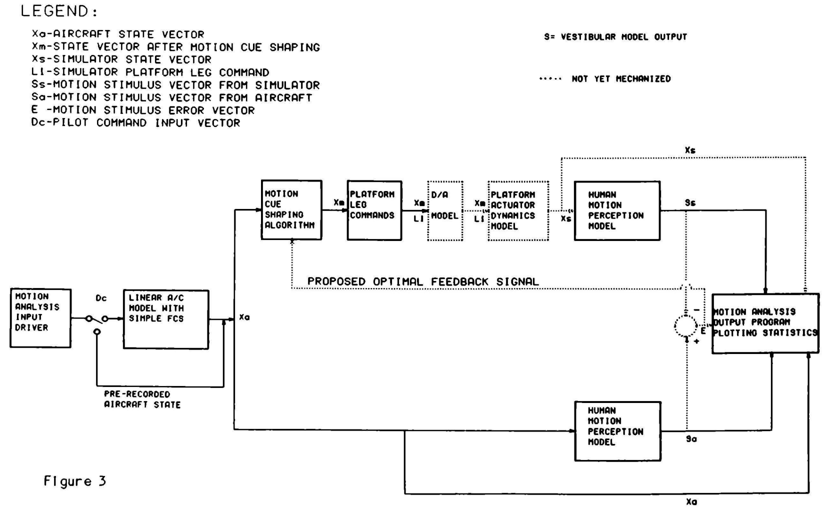

LEGEND:
Xa-AIRCRAFT STATE VECTOR
Xm-STATE VECTOR AFTER MOTION CUE SHAPING
Xs-SIMULATOR STATE VECTOR
Ll-SIMULATOR PLATFORM LEG COMMAND
Ss-MOTION STIMULUS VECTOR FROM SIMULATOR
Sa-MOTION STIMULUS VECTOR FROM AIRCRAFT
E -MOTION STIMULUS ERROR VECTOR
Dc-PILOT COMMAND INPUT VECTOR
S= VESTIBULAR MODEL OUTPUT
····· NOT YET MECHANIZED
MOTION ANALYSIS INPUT DRIVER
Dc
LINEAR A/C MODEL WITH SIMPLE FCS
PRE-RECORDED AIRCRAFT STATE
Xa
MOTION CUE SHAPING ALGORITHM
Xm
PLATFORM LEG COMMANDS
Xm
Ll
D/A MODEL
Xm
Ll
PLATFORM ACTUATOR DYNAMICS MODEL
Xs
Xs
HUMAN MOTION PERCEPTION MODEL
Ss
PROPOSED OPTIMAL FEEDBACK SIGNAL
-
E
+
MOTION ANALYSIS OUTPUT PROGRAM PLOTTING STATISTICS
HUMAN MOTION PERCEPTION MODEL
Sa
Xa

Figure 3

filter, and a non-linear coordinated adaptive washout filter. The motion cue shaping algorithm takes as its input either a direct input signal from the motion analysis input driver or state vector input from the aircraft math model. These input signals are filtered and shaped into simulator motion commands.

The simulator motion commands are sent to the third subsystem, the simulator platform hardware dynamics module, which is a mathematical model of the actual simulator motion hardware. The hardware model contains a zero-order-hold model simulating the digital to analog conversion and a transfer function representing the motion platform dynamic response.

The fourth subsystem, the human motion perception models, receives motion stimuli from either the filtered motion cue and motion hardware subsystems or directly from the aircraft parameter input module. There are two human motion perception sensors modeled: the otolith organs of the inner ear which are stimulated by linear accelerations, and the semicircular canals which respond to angular accelerations. The output of these perception models is perceived motion neuron afferent firing rate, which the brain processes to produce the sensation of motion. The afferent firing rates resulting from both the filtered simulator motion stimuli and the unfiltered aircraft motion stimuli can be analyzed and compared. Through this comparison, the difference between the filtered and unfiltered neuron firing rates can be minimized, either manually or through the proposed optimal feedback signal, producing simulator motion that the pilot perceives as realistic aircraft motion.

The final subsystem is a parameter plotting routine used to plot time histories of aircraft, motion cueing, and motion perception parameters for comparison.

The motion analytical tool is an interactive program that interfaces with the user by means of questions or messages printed on the user's computer terminal screen. The program requires input from the user in order to set up the time history input and output according to the user's wishes. It prompts the user for type and size of input, input entry time, and run duration time. As stated above, the user may select either the motion input driver or the simplified commercial transport aircraft math model to provide aircraft parameter input values.

Results

Two example cases are included to demonstrate the utility of the motion analysis tool. Notice that the sensed motion in the aircraft and in the simulator as illustrated by the afferent firing rates is the primary variable considered for comparison. The motion perception models act as nonlinear filters on the kinematic motion. Hence, it is not necessarily true that aircraft and simulator platform acceleration must correlate closely to achieve realistic motion cueing as long as the sensed motions between aircraft and simulator correlate well. The motion perceived by the brain is more significant to the pilot than the motion sensed by aircraft or simulator platform accelerometers.

The simulated aircraft for these examples is a simplified commercial wide-body transport type aircraft initialized at sea level with a velocity of 221 feet per second, in straight and level flight. The aircraft model, motion cueing algorithms and perception models were computed at a 50 Hz iteration rate. The first case is a longitudinal acceleration which stimulates primarily the otoliths. The second case is a roll maneuver intended to stimulate primarily the semicircular canals. The aircraft input consisted of a step input in thrust for the first case and a pulse input to the aileron surfaces for the second case. In both cases, the input was filtered through a first order lag to provide a more realistic aircraft response. The motion cueing algorithms used were nonlinear adaptive type (one of which is described in reference 8) set up to work with a hexapod, synergistic motion platform.

Figure 4 shows the response of the otolith model to a 15,000 pound aircraft thrust input. The time histories show (from bottom up) the increment of thrust from trim thrust, the aircraft longitudinal acceleration, the otolith afferent firing rate due to aircraft longitudinal acceleration and the otolith afferent firing rate due to simulator platform acceleration. Afferent firing rate is the neurological mechanism through which the human brain receives motion information. Observe that the simulator pilot would feel the initial acceleration cue after the aircraft pilot and sense a reverse acceleration cue before the pilot in the real aircraft. The strong acceleration onset cue appears to be partially caused by the breakout from the perception threshold. The sustained acceleration cue is not readily apparent because of the scaling of the plot. A gravity align method for generating sustained acceleration was not used in this particular case.

Figure 5 shows the response of the semicircular canal model to a 40 degree, four second aileron pulse. The time histories show (from bottom up) the aileron surface position, aircraft roll rate, aircraft roll acceleration, semicircular canal afferent firing rate due to aircraft roll acceleration and semicircular canal afferent firing rate due to simulator roll acceleration. Note that the simulator pilot perceives a roll sensation which is smaller in amplitude, occurs later and which exhibits a different shape than the roll sensation felt by the aircraft pilot.

Conclusions

The motion analysis program shows definite promise in separating qualitative and quantitative interpretations of simulator motion algorithms. With validated human perception models, the motion experimenter has the opportunity to isolate the particular motion parameters which make the greatest contributions to the pilot's perceived motion. Once identified, these parameters can be utilized to improve the quality of the simulator's motion system.

The analysis program is currently being used to investigate several motion algorithms. The culmination of this investigation will be an experiment designed to correlate subjective pilot opinions on motion sensations with motion sensations predicted by the human perception models.

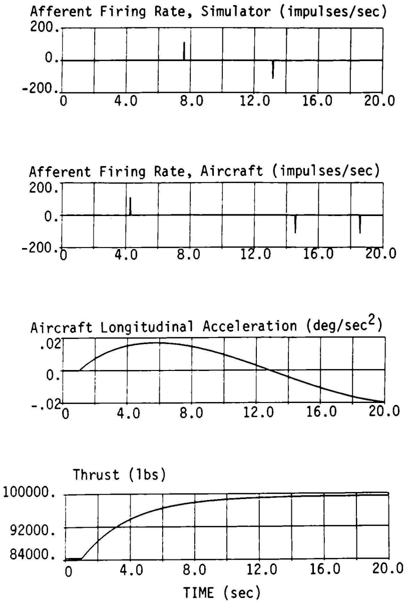


Figure 4 Otolith Response

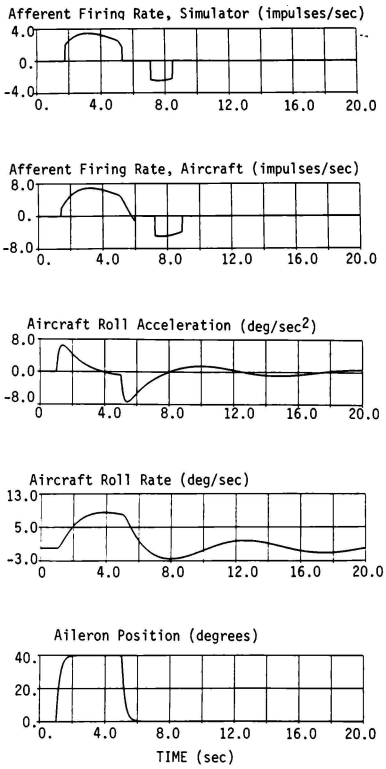


Figure 5 Semicircular Canal Response

References

1. Borah, J., L.R. Young, and R.E. Curry. "Sensory Mechanism Modeling," AFHRL-TR-77-70, (1977).

2. Borah, J., L.R. Young, R.E. Curry and W.B. Albery. "Multisensory Perception Model for Application to Aircraft Simulation," In the Proceedings for NAVTRAEQUICEN IH-306, pp. 85-92, (Nov 1978).

3. Cardullo, F.M., R.L. Kosut. "Old Problem/New Solutions: Motion Cueing Algorithms Revisited," In the Proceedings of 1983 AIAA Flight Simulation Technologies Conference, (Jan 1983).

4. Forsstrom, K.S. "An Investigation of Four Flight Simulator Motion Drive Algorithms," Unpublished, (1977).

5. Hayden, W.D. "Analytic Technique for Establishing the Motion Requirements for Ground Based Aircraft Flight Simulator," AIAA Paper 70-348, AIAA Visual and Motion Simulation Technology Conference, (1970).

6. Jaslow, H. "Human Tilt Perception in a Dynamic Environment," Aviation, Space and Environmental Medicine 50(6): 594-598, (1979).

7. Jaslow, H. "A Critique of the Gravity Vector Alignment Method for Motion Simulation," AIAA Paper 81-0985, (1981).

8. Martin, D.J. Jr. "A Digital Program for Motion Washout on Langley's Six-Degree-of-Freedom Motion Simulator," NASA Contractor Report 145219, (1977).

9. Ormsby, C.C. and L.R. Young. "Integration of Semicircular Canal and Otolith Information for Multisensory Orientation Stimuli," Mathematical Biosciences, 34:1-21, (1977).

10. Young, L.R. "Role of the Vestibular System in Posture and Movement," In: Medical Physiology, 13th Edition, V.B. Mountcastle (ed.), C.V. Mosby Co., St. Louis, MO., Vol. 1, pp. 704-721, (1974).

11. Young, L.R. "Perception of the Body in Space: Mechanisms," Handbook of Physiology--The Nervous System III, (Chapter 22), American Physiological Society, (ed. Ian Darian Smith), (1984).

85-1748

A SIMULATED INFRARED MODEL BOARD

C. Edward Jones*
James Lee**

Martin Marietta Aerospace
Orlando, Florida

Abstract

Martin Marietta Aerospace has built a model board that simulates infrared imagery for aircraft windscreen and sensor displays in its Simulation and Test Laboratory (STL). The simulation uses an image isocon TV camera with a Farrand optical probe; the video output is 525 or 875 line monochrome FLIR imagery. This paper outlines the design objectives, discusses the appearance of the simulated FLIR imagery on the displays, and describes the techniques used in constructing and painting the model board.

Introduction

Martin Marietta's Simulation and Test Laboratory is used for design, development, demonstration, and integration of avionics systems for fixed and rotary wing aircraft. Simulations of airborne mission equipment packages are performed for systems such as LANTIRN, TADS and LHX. The STL also demonstrates pilot and crew performance.

Man-in-the-loop simulation enables crew members seated in a cockpit and using visual cockpit displays to fly attack missions against targets positioned on an 80 by 40 foot terrain model.

The problem of present interest is to simulate a large infrared environment in a small scale. The temperature gradients existing in the real world are not easily reproduced in miniature; therefore, it is impractical to use a FLIR system in this simulation. Instead, target objects are painted to simulate FLIR characteristics so that a conventional black-and-white TV system displays targets and backgrounds in FLIR imagery.

Figure 1 shows the integration of the system elements. The 80 by 40 foot terrain model moves on tracks to simulate passage of terrain under the aircraft. The horizontal beam moves up and down to simulate changes in altitude, and the lateral carriage moves on the beam to simulate lateral movement. Two optical probes are mounted on the carriage. The front probe is part of a 50 degree optical system, with pitch, yaw, and roll that generates the aircraft windscreen view. The rear probe is a narrow field of view probe, independently controlled, that represents the aircraft fire control system.

Figure 2 shows the 80 by 40 foot horizontal model. Built on a fiberglass surface, it is strong enough to support personnel to make target changes. Mirrors three feet tall surround the terrain model to extend the visual terrain.

Three scales, 1200:1, 600:1, and 180:1, are incorporated on the model. The scales are designed not to interfere with one another.

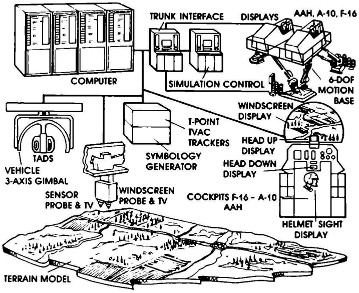


Figure 1. Integration of system elements.

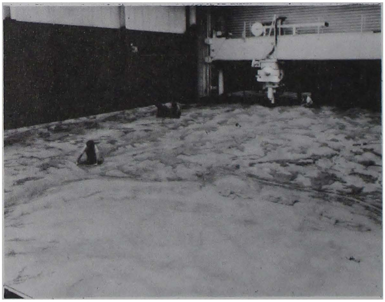


Figure 2. Terrain model.

Model Board Design Objectives

The design objectives were to:

1 Use a TV camera to create a 10.6 micron FLIR display for aircraft simulation.

2 Use the model board for rotary and fixed wing aircraft simulations.

3 Use altitudes as low as 15 feet (ground to pilot's eye level), for helicopter simulations. The 180:1 scale was chosen because the closest approach that the TV lens makes to the model is one inch, which represents 15 simulated feet to the pilot's eye.

*Manager, Simulation and Test Laboratory
**Engineering Prototype Staff Engineer

4 Have a long standoff capability. A 600:1 scale section is needed.

5 Have high-performance aircraft simulations. A 1200:1 scaling is also needed.

6 Have the whole model capable of being used with mixed scaling.

7 Have 10.6 micron FLIR video on all displays. The problem is that FLIR video has different display characteristics depending on the scene, season of the year, time of day, location, and many other variable conditions.

8 Have a large gaming area. Very few man-made objects would be included on the model so that a pilot trainee could not get familiar with the terrain.

9 Have the model easily changeable. Man-made objects and targets could be easily added or removed from the terrain. Objects such as power plants, Army depots, and tactical missions can be added easily.

10 Have more camera face plate illumination. The model would have lighter shades to get more light reflectance.

11 Have the model look as realistic as possible to the unaided eye. Trees need to be green, earth light olive, and roads light tan. FLIR characteristics would be made up of shades of color.

12 The model board would not be used for real FLIR sensor tests.

13 The sensor system does not simulate the noise and dead channels associated with FLIR systems.

FLIR TV Pictures

Ten-micron FLIR video is very scene-dependent, a function of many factors, such as sun angle, time after sunset, insulation of the material, cloud cover, cloud ceiling, absolute humidity, relative humidity, air temperature, sky radiance, wind, and water evaporation. Conditions for the construction of this model were chosen to give the characteristics of summer, late evening, and a remote location. A white hot condition is assumed for the TV system.

Data was collected from many video tapes of 10.6 micron FLIR made during low-level flight tests over the Fulda Gap area and from the deserts of the western USA.

Figure 3 is a photograph of a black hot 10 micron FLIR made during daylight hours. Figure 4 was taken at night. Trees and vegetation appear hot (black) when the sunlight is present, and neutral at night. The trunks and limbs are cool when shaded during the day. Roads and man-made objects often appear hot during the day and maintain heat for a long time after the sun goes down. In the early morning they appear cold. Operating vehicles appear hot around the engine compartment. Life forms (people and animals) appear hot. Water varies greatly with reflected light, although reflections appear darker (reflecting the sky) than an actual object seen in the water. The water effect can be achieved on the model by using a smooth surface that reflects light.

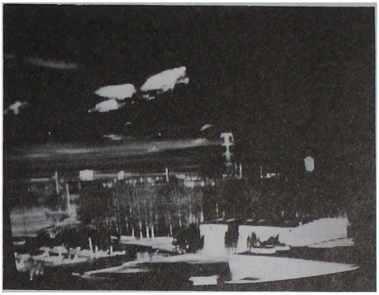


Figure 3. Black hot FLIR photograph taken during daylight.

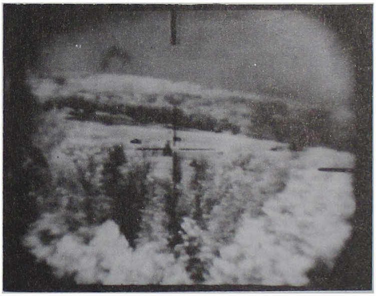


Figure 4. Black hot FLIR photograph taken at night.

Model Construction

Figure 5 is a white hot thermal photograph of an oil tank and city surroundings. The difference in temperature at the top of the tank is caused by the fact that the tank is half full. Figure 6 is the scale model with the tanks painted lighter at the top to achieve the same effect. Figure 7 is the TV monitor display of a simulated FLIR.

Vegetation

Trees are the most noticeable feature and have the greatest overall visual effect of any of the model's features. They are also the most

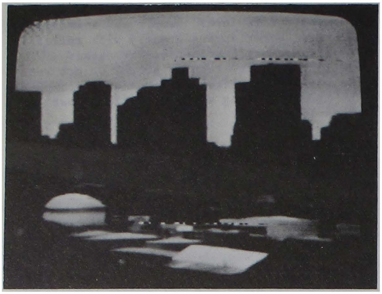


Figure 5. Thermal photograph of oil tank in city.

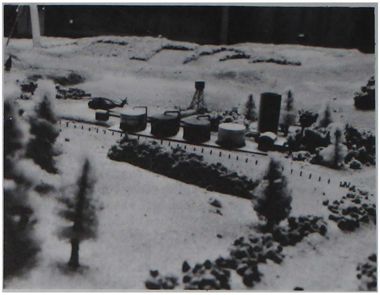


Figure 6. Scale model.

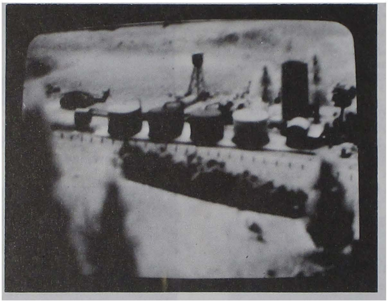


Figure 7. TV display of simulated FLIR.

difficult feature to model, but are most important to conveying the overall feeling of realism, orientation, altitude and scale. The most obvious characteristic of trees with respect to FLIR is that only the main branches and trunk are visible, with a fuzzy, somewhat blurred or ghostly image representing branch and leaf areas. When leaves are present (summer) they generally contrast about 5 to 10 percent darker than the surrounding ground area. In winter, trees show high contrast against snow backgrounds; no ghost effects are present due to absence of foilage.

Trees in groups show detail to about the second row only, with the leaf effects eventually blotting out trees to the rear. The approach to the tree effect was on three levels:

1 High-detail individual trees.

2 Medium detail with trees in groups and for edges of all groups of trees along the flight corridors.

3 Low detail tree tops only to represent large groups of trees. Foam rubber, finely chopped, painted, and glued in place was used. Other techniques include vacuum formed and electrostatically flocked shapes to give a pine tree effect, molded shapes with flocking, and horse hair packing material with fine foam rubber sprinkled and held on with glue.

Ground Texture

Ground texture was created with combinations of sand of various grit types mixed into the painted surface. The value of ground reflectance appears to be in the neutral range, 45 to 55 degrees. Generally, the ground surface is slightly cooler than the surrounding environment which results in ground appearing slightly lighter than average in black hot imagery and slightly darker in white hot imagery.

Roads

Roads appear lighter than the surrounding ground. In general, roads appear near neutral range.

Tall Vertical Structures

Telephone poles and high tension tower models are metal etched, and painted lighter to represent hotter surfaces. Wires, which are hot, are usually lighter than their backgrounds. Objects such as transformers are hot (painted lighter), and water towers and oil tanks, which have large heat sinks, appear cold, and the models are painted darker.

Buildings

Buildings are fabricated to scale and show all detail. In cold weather, the outsides of the buildings appear darker than average, in the 70 to 80 percent range. The insides of open sheds appeared darker than the outsides because of the shaded areas.

Water

Water reflects light in various ways. Reflections appear darker than the actual object reflected. Water is detected mainly by its texture when smooth and by high contrast levels and patterns. In simulation, water effect is achieved generally by using a smooth surface effect, since small bodies of water are most likely all that is required. Assuming summer, water should be colder than ground and thus darker. Reflected light from the sky could actually make water appear warmer and thus lighter in certain areas.

Vehicles

Vehicles are molded, cast in epoxy, and painted with enough detail to be easily recognized. Features such as wheels, turrets, machine guns, rockets, and gas tanks are shown. Heat generated by warm engines, treads and wheels is detailed by painting these areas lighter. The contrast range on a running tank is very high, 30 to 90 percent. Tanks, and similar vehicles in and around trees, can be easily detected by their contrast levels.

Figure 8 shows a chart of 6 by 6 inch paint chips of various model color shades. The chart is used as a measuring device to check values and contrast levels. Since it had been determined that the overall light level of the model should be increased and that light and dark were actually relative to a gray scale, the chart was developed to compare various colors to their corresponding gray value. The relative video signal level from each chip was measured to determine the colors to be used. All chips were measured at the same angle. A latex paint was chosen and a color chart from the vendor was used as a reference for the percentage of light reflected. Since the lens system for the camera needs light, beige was chosen so that the background is in the middle of the reflectance range. This gives the latitude to make objects very much hotter or much colder than their environment. A FLIR is often adjusted to the background or average ambient temperature. The beige was tinted to dark green, light green and brown and textured with sand of 30-65 grit to achieve the required earth tones and surface textures.

The Color Value Chart

Figure 9 shows the color chart used as a measuring device to check contrast levels. The chart has 16 gray values plus black and white. The objective was to find more than one color to fit a given contrast. For each gray chip, four separate hues were matched for equal contrast level. The color chips were from a Sherwin Williams "Beau Monde" collection. These samples were listed by percentage of light reflected value (LRV). A five percent for dark blue to 85 percent for off white. A gray having a LRV of 26 percent could be matched with blue, red, green or brown. Many colors can be used to represent a shade of gray on the model.

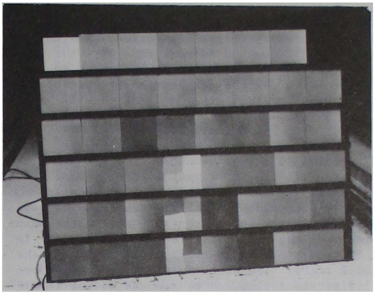


Figure 8. Color chart.

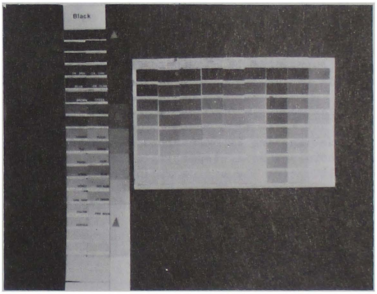


Figure 9. Color chart used to check contrast levels.

To the naked eye, colors in the 40 to 60 percent LRV region show best separation of contrasts, while darker or lighter colors are not vivid, nor do they represent natural earth tones.

A color also appears to change in value relative to the background. A red diamond chip with an LRV of 30 percent disappears against a gray chip of 30 percent. To the eye (or looking at a TV monitor) the 30 percent red diamond chip appears dark against a 10 percent background. The same chip appears almost white against a 60 percent background. This effect must be considered for small targets against a broad background.

Effects of Camera Optics

The optical probe is a sophisticated lens system for the TV cameras. Because of the number of lens surfaces and mirror surfaces (some not so perfect), light scatters and causes whites to spill into the adjacent black region. The result is a gray scale without the 0 to 10 or 80 percent and above LRV.

To evaluate the needed contrast levels, a sample was painted and observations were made through the probe. The FLIR TV was compared with the probe TV and sections were repainted. High, low and medium contrasts were tried. Medium contrast was found to be realistic with the FLIR TV. The resulting video was evaluted on an oscilloscope to check video level and contrast against that of the FLIR.

Overlays

Figure 10 shows the model with the 180:1 scale terrain around the outer edges. Mountain ranges with passes separate the 180:1 scales. The 1200:1 sections are in the center. When the optical probe is in the central section the mountains hide the 180:1 scale and when the probe is at the edges of the model the mountains hide the 1200:1 scale. The overlay is divided into three rigid sections that can be removed within thirty minutes. These overlays cover the 1200:1 scale with 600:1 scale terrain.

Because of the mountain ranges and passes, flight over the terrain can be chosen so that an endless variety of routes is available. A pilot need not quickly become familiar with the terrain. Flights can be chosen to follow a large number of routes around the terrain.

Model Scaling

Model scaling is determined from such operating conditions as altitude, speed and range. Some helicopters fly as low as 15 feet (eye level above the surface). The probe will go as low as one inch above the model, which indicates a scale of 180:1. Because some target ranges approach 15,000 feet, some areas of the model board are occasionally used for such high performance fixed wing aircraft as the F-16. With the speed and turn radius of an F-16 a larger area is needed, so the center portion of the model is scaled at 1200:1.

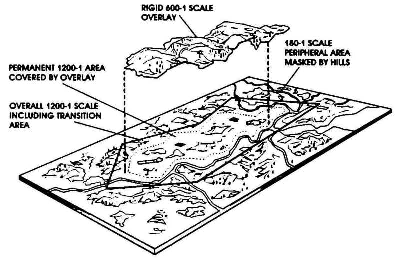

RIGID 600-1 SCALE OVERLAY
180-1 SCALE PERIPHERAL AREA MASKED BY HILLS
PERMANENT 1200-1 AREA COVERED BY OVERLAY
OVERALL 1200-1 SCALE INCLUDING TRANSITION AREA

Figure 10. Terrain model, overlay, and scaling.

The transition from one scale to another need not be abrupt. The transition can be made through nondescript vegetation that can represent anything from bushes to trees. When the pilot flies through the transition area he need not speed up or slow down. Missions are planned to stay in specific scaled areas.

Conclusions

The color values chosen for the model were selected by an artist experienced with FLIR TV scenes. The artist compared FLIR TV displays with the scenes coming from the model board. He then applied the colors corresponding to the thermal contrast. This approach provides a realistic FLIR TV presentation for man-in-the-loop aircraft simulations.

85-1749

A RING-VORTEX DOWNBURST MODEL FOR REAL-TIME FLIGHT SIMULATION OF SEVERE WIND SHEARS

Michael Ivan*
Boeing Computer Services Company
Seattle, Washington

Abstract

A mathematical model of a downburst was developed to be suitable for real-time flight simulations of take-offs and landings in low-altitude severe wind-shears due to downbursts or microbursts with circulatory wind-flow patterns resembling those noted in downburst data from the Joint Airport Weather Studies (JAWS) Project. The mature stage of downburst wind-flow patterns was idealized as a three-dimensional axisymmetric circulatory flow field similar to that around a horizontal smoke-ring or ring-vortex at an appropriate height above the ground. The flow field around a ring-vortex has a stream function expressed in the classical textbook "Hydrodynamics" in terms of the complete elliptic integrals combination of [F(k) - E(k)]; which here is approximated by the expression: $[(.788k^2/(.25+.75\sqrt{1.-k^2})]$, in the limited range of $0. \le k^2 \le 1.$ for the modulus $k=(r_2-r_1)/(r_2+r_1)$, where r_1 and r_2 denote the least and greatest distances, respectively, of the point P from the ring vortex. "Digital differentiations" of the downburst stream function yield both the WZ downdraft velocity component and the WR radial wind velocity component at the airplane center-of-gravity position under the ring vortex. The WR radial velocity component is then resolved into the two horizontal components WX and WY for wind speeds along and across the runway, respectively. Occupying 383 words of memory and having an average 1.3 millisecond real-time execution on a Harris H800 digital computer, the present ring-vortex downburst model provides economical simulation of severe wind-shear flow patterns that resemble closely some of the flow patterns noted in meterological data from the JAWS Project.

Nomenclature (in Eq↓)

ACMPEM	Approx Comp Elliptic Integrals Mir R-V† (13)
ACMPEP	Approx Comp Elliptic Integrals Pri R-V (11)
CIRCRV	Circulation Strength of Primary R-Vortex (1)
GSX	X-Coord. Along Runway From Threshold, Feet
GSY	Y-Coordinate From Runway Centerline, Feet
H	Barometric Altitude of Airplane CG, Feet
HGCG	Center-of-Gravity Altitude Above Ground, Ft
HGROUN	Altitude of Ground Level Below CG, Feet
HGVORT	Height Above Ground to Primary R-Vortex, Ft
KMODMV	K Modulus for Elliptic I. Mir R-Vortex (12)
KMODPV	K Modulus for Elliptic I. Pri R-Vortex (10)
PSIRW	Orientation Angle of Runway From North, Deg
RCOHGV	Ratio of Core Radius to Ref. Height, HGVORT
RDCORE	Radius of Core of Rotating Vortex Material
RDVORT	Reference Radius of Ring-Vortex, Feet
RVAXCG	Radius to CG From Vertical Axis of Dnb. (5)
R1MSML	Small Distance "R1" From CG to Mir. R-V (7)
R1PSML	Small Distance "R1" From CG to Prim R-V (6)
R2MLRG	Large Distance "R2" From CG to Mir. R-V (9)
R2PLRG	Large Distance "R2" From CG to Prim R-V (8)
STRFCG	Stream Function at CG From R-Vortex Db (18)
STRFCR	Stream Function at Unit Radial Incr CG (19)
STRFCZ	Stream Function at Unit "Z" Incr fr CG (17)
STRMFN	Stream Function at CG From R-Vortex Db (14)
SX	CG Displacement North From Threshold of R/W
SY	CG Displacement East From Threshold of R/W
WR	Radial Wind Speed of Outflow From Axis (15)
WX	Horizontal Tail-Wind Comp in X-R/W Dir (20)
WY	Horizontal Cross-Wind Comp. Y-R/W Dir. (21)
WZ	Downdraft Wind Comp. Down "Z" Dir. (2, 16)
WZREF	Reference Axial Downdraft at Height HGVORT
XCGDBC	X-Runway Distance of CG From Db Center (3)
XRWDBC	X-Runway Coordinate to Db Central Axis
YCGDBC	Y-Runway Distance of CG From Db Center (4)
YRWDBC	Y-Runway Coordinate to Db Central Axis

1. Introduction

A variety of weather features such as mountain lee waves, thunderstorms, frontal shears, and downbursts have been tabulated in [1] as probable causes of 27 hazardous encounters with low-altitude severe wind-shears. However, here attention will be focused only on the downburst wind-shear phenomenon because this single weather feature was the probable cause of approximately half of those hazardous encounters. Usually associated with a thunderstorm, "downburst" and "microburst" are used here interchangeably to denote a low-altitude severe wind-shear flow pattern defined in [2] as: "a downdraft induced, diverging, horizontal flow near the surface, whose initial horizontal dimension is less than 4 kilometers, and whose differential velocity is greater than 10 meters per second."

The distinctive label of "microbursts" had been given to downbursts with wind-shears greater than 10 m/s across diameters in the size range of 1 to 4 kilometers in [3] and [4] where meteorological analyses indicated a greatest probability of severe wind-shears from mature downbursts with particularly critical diameters of approximately 3 kilometers or the length of a 10,000 foot runway as shown below the Figure 1 wind-flow diagram. This mature outburst stage occurs after a downdraft has impinged against the ground and been deflected initially into a horizontal radial outburst flow that later curls up from the ground to form the mature downburst flow pattern shown in Figure 1. Circulatory wind-flow patterns similar to those around a horizontal "smoke-ring" or ring-vortex at an appropriate height above the ground had been noted in downburst wind-shear data from the Joint Airport Weather Studies (JAWS) Project in [2] and [5]. Such microbursts can form rapidly to full intensity in less than five minutes and usually have a short duration of less than ten minutes at full intensity.

Real-time flight simulations were being used to study possible solutions to the problems of air-

†R-V and R-Vortex: Ring-Vortex

*Senior Engineer, Simulation Applications Software, Member AIAA.

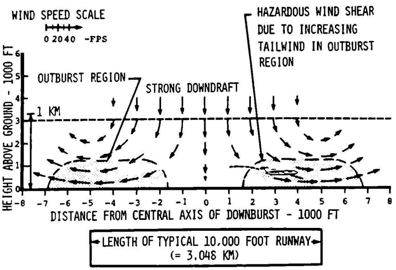

WIND SPEED SCALE
0 20 40 -FPS
HAZARDOUS WIND SHEAR DUE TO INCREASING TAILWIND IN OUTBURST REGION
OUTBURST REGION
STRONG DOWNDRAFT
1 KM
HEIGHT ABOVE GROUND - 1000 FT
0
1
2
3
4
5
6
-8
-7
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
7
8
DISTANCE FROM CENTRAL AXIS OF DOWNBURST - 1000 FT
LENGTH OF TYPICAL 10,000 FOOT RUNWAY
(= 3,048 KM)

FIGURE 1 : WIND FLOW PATTERN IN A RADIAL CROSS-SECTION THROUGH A TYPICAL MICROBURST AS A HAZARDOUS MATURE DOWNBURST

plane flights through severe wind shears, and these simulations required a downburst mathematical model that would be both economical with computer resources and versatile enough to simulate a range of important parameters such as size, shape, and intensity of the downburst model. Only the low-altitude wind-flow regions within, say, 500 feet of the ground are of primary concern for physically realistic simulations of severe wind-shears because these low-altitude winds will have the most critical effects on the flight path trajectory of an airplane relative to the ground. An initial candidate for a fairly realistic simulation of the above mature downburst model was a three-dimensional axi-symmetric flow field developed [6] between two circular horizontal doublet-sheets with suitable radial distributions of the singularity intensity. The primary doublet-sheet above ground level was matched below ground level by a mirror image second doublet-sheet with opposite sign to satisfy the boundary condition for only horizontal flow velocity along the ground level.

In [6] the major attention was focused on a downburst model based on a cosine radial distribution of the doublet singularity intensity, and that cosine distribution required extensive integrations to calculate the resulting flow field. However, some flow data were shown in [6] also for a simple downburst model based on a uniform distribution of the doublet-sheet singularity. It was noted that the flow field from such a uniform doublet distribution corresponded also to that from two ring-vortices in place of the two circular edges of the doublet sheets. The present downburst model is thus based on the simple concept of two ring-vortices producing a flow field that could be calculated easily in a "closed-form solution" not requiring extensive integrations.

2. Stream function for Ring-Vortex Downburst

The simplicity of the latter downburst model based on two ring-vortices was enhanced by the use of the classical "closed-form" equations (11) and (12) on page 237 in [7] for the stream function for the flow field around an isolated ring vortex. A constant value of the stream function will define the coordinates along a steady streamline in that flow field even though the streamline pattern in a downburst may be only of academic interest. However, the downburst's velocity components are of prime importance for flight simulations and will be calculated later from the stream function by appropriate differentiations per equation (1) on page 236 in [7] to yield the radial and axial velocity components.

The ring-vortex stream function in the above equation (11) is expressed in terms of the following combination of the complete elliptic integrals: $[F(k) - E(k)]$. These "higher functions" are described on pages 43-61 in [8]. The present model calculates the downburst stream function at a point P by approximating the combination $[F(k) - E(k)]$ with the expression: $[(.788k^2)/(.25+.75\sqrt{1.-k^2})]$ in the limited range of $0. \leq k^2 \leq 1.$ for the modulus $k=(r_2-r_1)/(r_2+r_1)$, where r_1 and r_2 denote the least and greatest distances, respectively, of the point P from the ring vortex.

Figure 2 represents an oblique over-view of a ring-vortex downburst model positioned in runway-oriented coordinates to the right of the takeoff end of a runway. Figure 3 presents an enlarged oblique view of the ring-vortex downburst model to show some of the geometric details that enter into the calculations of first the stream function STRMFN, and then the runway-oriented wind-shear velocity components WX, WY, and WZ.

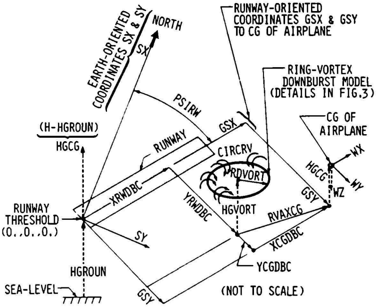

EARTH-ORIENTED COORDINATES SX & SY
NORTH
SX
RUNWAY-ORIENTED COORDINATES GSX & GSY TO CG OF AIRPLANE
RING-VORTEX DOWNBURST MODEL (DETAILS IN FIG.3)
PSIRW
GSX
CG OF AIRPLANE
(H-HGROUN)
HGCG
RUNWAY
CIRCRV
RDVORT
WX
HGCG
WY
WZ
GSY
XRWDBC
YRWDBC
HGVORT
RVAXCG
RUNWAY THRESHOLD (0.,0.,0.)
SY
XCGDBC
YCGDBC
HGROUN
SEA-LEVEL
GSY
(NOT TO SCALE)

FIG. 2 OBLIQUE OVER-VIEW OF RING-VORTEX DOWNBURST MODEL

Default data values will be enclosed in parentheses to indicate appropriate typical values and units for some of the input parameters used to define a downburst model classified as producing medium intensity windshears. The downburst model shown in Figures 2 and 3 is based on a horizontal ring-vortex with a reference radius RDVORT (=5000. ft.) located above the ground level at a reference height of HGVORT (=3000. ft.) at which altitude a reference downdraft velocity of WZREF(=35. fps) will flow down the central axis. Either doubling or halving the WZREF reference downdraft velocity will proportionately either increase the windshears to a high intensity category or decrease the windshears to a low intensity category, respectively.

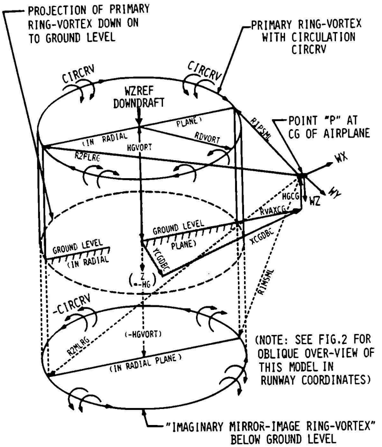

PROJECTION OF PRIMARY RING-VORTEX DOWN ON TO GROUND LEVEL
PRIMARY RING-VORTEX WITH CIRCULATION CIRCRV
CIRCRV
CIRCRV
WZREF DOWNDRAFT
(IN RADIAL PLANE)
RDVORT
HGVORT
R2PLRG
R1PSML
POINT "P" AT CG OF AIRPLANE
WX
WY
WZ
HGCG
RVAXCG
GROUND LEVEL (IN RADIAL PLANE)
GROUND LEVEL (IN RADIAL
XCGDBC
YCGDBC
(Z = -HG)
R1MSML
-CIRCRV
(-HGVORT)
R2MLRG
(IN RADIAL PLANE)
(NOTE: SEE FIG.2 FOR OBLIQUE OVER-VIEW OF THIS MODEL IN RUNWAY COORDINATES)
"IMAGINARY MIRROR-IMAGE RING-VORTEX" BELOW GROUND LEVEL

FIG.3 GEOMETRIC DETAILS OF RING-VORTEX DOWNBURST MODEL

The velocities in this downburst model are proportional to the circulation strength of the primary ring-vortex, CIRCRV, which is related to the WZREF reference downdraft through the following equation:

$$\mathrm{CIRCRV} = \frac{\mathrm{WZREF} \;*\; (2.*\mathrm{RDVORT})}{\left\{ \underbrace{1.}_{\text{"PRIMARY RING-VORTEX EFFECT"}} \quad \underbrace{\frac{-1.}{\left[1.+\left(\frac{2.*\mathrm{HGVORT}}{\mathrm{RDVORT}}\right)^{2}\right]^{1.5}}}_{\text{"MIRROR-IMAGE EFFECT"}} \right\}} \qquad (1)$$

The above equation shows both the effect of the "imaginary mirror-image ring-vortex", assumed to be below ground level, and the effect of the "primary ring-vortex" above ground level. The above equation is based on an equation presented on page 93 in [9], in relation to the Figure 2.13.5, shown there for "velocity induced by a circular vortex ring at a point on the axis". This latter equation was also used to develop the following equation (2) for the WZ downdraft velocity down the vertical central axis of the downburst model:

$$\mathrm{WZ} = \left(\frac{\mathrm{CIRCRV}}{2.*\mathrm{RDVORT}}\right) * \left\{ \underbrace{\frac{1.}{\left[1.+\left(\frac{\mathrm{HGVORT}-\mathrm{HGCG}}{\mathrm{RDVORT}}\right)^{2}\right]^{1.5}}}_{\text{"PRIMARY RING-VORTEX EFFECT"}} \quad \underbrace{\frac{-1.}{\left[1.+\left(\frac{-\mathrm{HGVORT}-\mathrm{HGCG}}{\mathrm{RDVORT}}\right)^{2}\right]^{1.5}}}_{\text{"MIRROR-IMAGE EFFECT"}} \right\} \qquad (2)$$

As expected, the equation (2) downdraft velocity diminishes to zero at the stagnation point at ground level where HGCG=0.

The ring-vortex downburst model has radial symmetry about its vertical central axis, and shown on Figure 2 is the radial distance, RVAXCG, calculated as follows from that vertical axis to the CG of the airplane:

$$\mathrm{XCGDBC} = \mathrm{GSX} - \mathrm{XRWDBC} \qquad (3)$$

$$\mathrm{YCGDBC} = \mathrm{GSY} - \mathrm{YRWDBC} \qquad (4)$$

$$\mathrm{RVAXCG} = \sqrt{(\mathrm{XCGDBC})^{2} + (\mathrm{YCGDBC})^{2}} \qquad (5)$$

In the radial plane that contains RVAXCG, there are also the following four important distances measured from the CG as shown in Figure 3:

R1PSML and R1MSML as the small (least) distances "R1" from CG to, respectively, the "primary" and the "mirror-image" ring vortices.

$$\mathrm{R1PSML} = \sqrt{(\mathrm{HGCG}-\mathrm{HGVORT})^{2} + (\mathrm{RVAXCG}-\mathrm{RDVORT})^{2}} \qquad (6)$$

$$\mathrm{R1MSML} = \sqrt{(\mathrm{HGCG}+\mathrm{HGVORT})^{2} + (\mathrm{RVAXCG}-\mathrm{RDVORT})^{2}} \qquad (7)$$

R2PLRG and R2MLRG as the large (greatest) distances "R2" from CG to, respectively, the "primary" and the "mirror-image" ring vortices:

$$\mathrm{R2PLRG} = \sqrt{(\mathrm{HGCG}-\mathrm{HGVORT})^{2} + (\mathrm{RVAXCG}+\mathrm{RDVORT})^{2}} \qquad (8)$$

$$\mathrm{R2MLRG} = \sqrt{(\mathrm{HGCG}+\mathrm{HGVORT})^{2} + (\mathrm{RVAXCG}+\mathrm{RDVORT})^{2}} \qquad (9)$$

The preceding "R1" and "R2" distances to the "primary" ring-vortex were used to calculate the KMODPV as the corresponding "k modulus" argument for the following ACMPEP "approximation for the complete elliptic integrals applicable to the primary ring vortex":

$$\mathrm{KMODPV} = (\mathrm{R2PLRG}-\mathrm{R1PSML})/(\mathrm{R2PLRG}+\mathrm{R1PSML}) \qquad (10)$$

$$\mathrm{ACMPEP} = (.788*\mathrm{KMODPV}^{2})/(.25+.75\sqrt{1.-\mathrm{KMODPV}^{2}}) \qquad (11)$$

Similarly, the following corresponding KMODMV and ACMPEM were calculated for the "mirror-image ring vortex":

$$\mathrm{KMODMV} = (\mathrm{R2MLRG}-\mathrm{R1MSML})/(\mathrm{R2MLRG}+\mathrm{R1MSML}) \qquad (12)$$

$$\mathrm{ACMPEM} = (.788*\mathrm{KMODMV}^{2})/(.25+.75\sqrt{1.-\mathrm{KMODMV}^{2}}) \qquad (13)$$

Finally, the STRMFN stream function at the CG of the airplane is:

$$\mathrm{STRMFN} = \left(\frac{-\mathrm{CIRCRV}}{2\pi}\right) * \left[\begin{array}{l} (\mathrm{R1PSML}+\mathrm{R2PLRG})*\mathrm{ACMPEP} \\ -(\mathrm{R1MSML}+\mathrm{R2MLRG})*\mathrm{ACMPEM} \end{array}\right] \qquad (14)$$

3. The Velocity Components of Wind Shear

On the basis of equation (1) on page 236 in [7], the WR radial velocity component and the WZ downdraft velocity component are related to the following mathematical derivatives and approximate "digital differentiations" of the preceding STRMFN stream function outside of the unit radius of RVAXCG:

$$WR = \frac{1}{RVAXCG} * \left[\frac{d(STRMFN)}{d(Z)}\right] \simeq \frac{(STRFCZ - STRFCG)}{RVAXCG} \quad (15)$$

$$WZ = \frac{-1}{RVAXCG} * \left[\frac{d(STRMFN)}{d(RVAXCG)}\right] \simeq \frac{(STRFCG - STRFCR)}{RVAXCG} \quad (16)$$

Where: STRFCZ=STRMFN ((HGCG-1.), RVAXCG) with a unit "Z" displacement applied as a unit decrement to HGCG to yield (HGCG-1.) as the revised argument for STRMFN (equation 14). (17)

STRFCG=STRMFN(HGCG, RVAXCG) at the CG of airplane. (18)

STRFCR=STRMFN(HGCG, (RVAXCG+1.)) with a unit "Radial" displacement applied as a unit increment to RVAXCG. (19)

Still outside of the unit radius of RVAXCG, the above WR radial component is projected (using results from equations 3-5) into the following two horizontal components WX and WY in runway-oriented coordinates:

WX = WR * XCGDBC/RVAXCG ______________ (20)
WY = WR * YCGDBC/RVAXCG ______________ (21)

The preceding equations (6)-(21) were arbitrarily restricted to apply outside of the unit radius of RVAXCG to avoid possible attempted division by zero as RVAXCG approached zero at the vertical central axis of this downburst model.

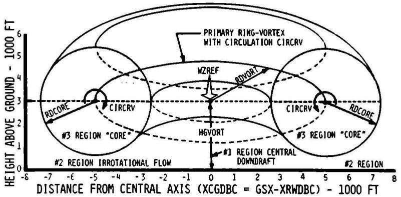
However, within the unit radius of RVAXCG, there is arbitrary setting to zero of the WR radial components and also both horizontal components WX and WY. In this unit radius cylindrical region, only the remaining WZ downdraft component is calculated to be the "simple" central axial downdraft based on the preceding equation (2).

The label of "#1 Region Central Downdraft" is applied in Figure 4A to the unit radius cylindrical region around the central vertical axis, and also shown in Figure 4A are the other two regions labeled as:

"#3 Region Core" is an assumed circular toroid of "core vortex material" executing rotational flow similar to rigid-body-rotation.

"#2 Region Irrotational Flow" is the induced irrotational flow region remaining outside of both of the preceding #3 Region and #1 Region.

The concept behind the rigid-body-rotations inside the #3 Region Core is to calculate physically realistic velocity components through a fairly simple interpolation scheme in order to avoid the theoretically large velocities calculated near a concentrated vortex. Beginning from an arbitrary zero velocity at the center of each circular cross-section of this "core", the WR and WZ velocity components are increased linearly as indicated in Figure 4B to match the velocity components calculated on the "core" surface at a cross-sectional "core radius" RDCORE. The RDCORE is assumed to be proportional to the reference height, HGVORT, through the ratio RCOHGV, equal to 0.8 in default data.



PRIMARY RING-VORTEX WITH CIRCULATION CIRCRV
WZREF
RDVORT
RDCORE
CIRCRV
CIRCRV
RDCORE
#3 REGION "CORE"
HGVORT
#3 REGION "CORE"
#1 REGION CENTRAL DOWNDRAFT
#2 REGION IRROTATIONAL FLOW
#2 REGION
HEIGHT ABOVE GROUND - 1000 FT
0
1
2
3
4
5
6
-8
-7
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
7
8
DISTANCE FROM CENTRAL AXIS (XCGDBC = GSX-XRWDBC) - 1000 FT

FIGURE 4A : THE THREE REGIONS OF WIND-SHEARS IN THE IVAN RING-VORTEX DOWNBURST MODEL

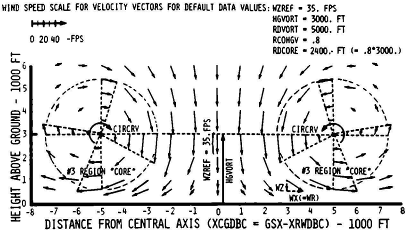


FIGURE 4B: TYPICAL WIND-SHEAR VELOCITY VECTORS IN A RADIAL CROSS-SECTION THROUGH DOWNBURST MODEL

4. Sample Flow Patterns from Downburst Model

The sample idealized downburst model shown in Figure 4B is based on wind-shear velocity components WX and WZ calculated by a digital computer program using the noted default data values to define the size and intensity of the wind-shear model. Those data values had been selected to simulate a medium intensity wind-shear level with a change in horizontal wind speed of 82 fps (=25 M/S) occurring along a 10,000 foot diameter of the downburst model. Downbursts with diameters in that general size of 10,000 feet or three kilometers were indicated in [1] through [4] to be the critical sizes that are hazardous to transport airplanes during takeoffs and landings.

The digital computer program for this downburst model was written in the FORTRAN 77 programming language and occupied 383 words of memory in a Harris H800 digital computer. In real-time flight simulations on that computer, this downburst model had an average execution time of 1.3 milliseconds.

Figure 5A shows a downburst flow pattern in a vertical cross-section of wind components obtained from Joint Airport Weather Studies (JAWS), 5 August 1982 wind-shear data [5], and a similar flow pattern was also shown in the 14 July 1982 JAWS wind-shear data presented in Figure 2(b) in [2]. Figure 5B shows calculated wind-shear velocities from the present downburst model with the noted input parameters to approximate the Figure 5A downburst flow pattern. Comparison of Figure 5B with Figure 5A indicates that the present downburst model can simulate wind flow patterns that are physically realistic approximations of some of the major wind flow patterns obtained from JAWS 5 August 1982 wind-shear data.

This downburst model has been implemented and used successfully in the Engineering Simulation Center at Boeing.

5. Conclusion

Three features of the present version of a ring-vortex model downburst make it suitable for use in real-time simulations of airplane flights through downbursts near the ground:

1. Physically realistic approximations of some JAWS wind-shear patterns are calculated by this idealized downburst model.

2. Flexibility in varying several input parameters enables this model to simulate a wide range of wind-shear intensity levels and sizes of downbursts.

3. Economical uses of real-time computer resources are illustrated by this model through an average execution time of 1.3 milliseconds while occupying 383 words of memory in a Harris H800 digital computer.

6. Acknowledgment

In the development of this paper, I gratefully appreciate the encouragements from John Potter and Keith Hill, and I thank Julie Fulmer for typing the manuscript and David Babcock and Dori Wysocki for the computer calculations and preparation of the diagrams.

REFERENCES

1. National Research Council: Low-Altitude Wind Shear and Its Hazard to Aviation, 1983, National Academy Press, Washington, D.C., pages 14-15.

2. McCarthy, J., and Elmore, K.: JAWS Analysis Highlights in the Aviation Safety Context, Preprint, 29th Corporate Aviation Safety Seminar, Montreal, Quebec, Canada, April 1-3, 1984, 24 pp.

3. Fujita, T. T.: Microbursts as an Aviation Wind Shear Hazard, Preprint AIAA-81-0386, AIAA 19th Aerospace Sciences Meeting, St. Louis, Missouri, January 12-15, 1981, 9 pp.

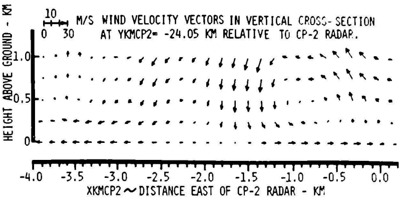


FIGURE 5A : JAWS 5 AUG. 1982 WIND-SHEAR DATA SHOWING DOWNBURST FLOW PATTERN

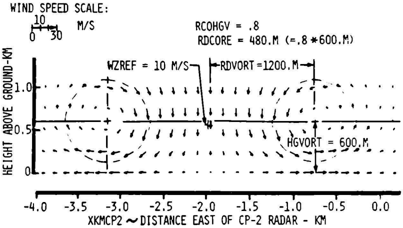


FIGURE 5B : IDEALIZED DOWNBURST MODEL APPROXIMATION OF THE ABOVE FIGURE 5A JAWS DATA

4. Fujita, T. T.: Tornadoes and Downbursts in the Context of Generalized Planetary Scales, Journal of Atmospheric Sciences, Vol. 38, No. 8, August 1981, pp. 1511-1534.

5. National Center for Atmospheric Research: "The JAWS Project Preliminary Data Description", (for the 5 August 1982, 1847 MDT Wind Shear Data near Stapleton International Airport, Denver, Colorado), Boulder, Colorado, September 1983, 102 pp.

6. Zhu, S., and Etkin, B. E.: "Fluid-Dynamic Model of a Downburst", University of Toronto, UTIAS Report No. 271, April 1983, 46 pp.

7. Lamb, Sir Horace: "Hydrodynamics", 1932 Sixth Edition, New York, Dover Publications, Inc., pages 236-237.

8. Losch, F.: "Jahnke-Emde-Losch Tables of Higher Functions", 1960 Sixth Edition, New York, McGraw-Hill Book Company, Inc., pages 43-61.

9. Duncan, W. J., Thom, A. S., and Young, D. A.: "Mechanics of Fluids", 1970 Second Edition, New York, American Elsevier Publishing Company, Inc., pages 89-94.

85-1750

COMPOSITE STATISTICAL METHOD FOR MODELING WIND GUSTS FOR AIRCRAFT SIMULATION

James R. Schiess*
NASA Langley Research Center
Hampton, VA 23665

Abstract

This paper discusses the application of three statistical methods in combination to model wind gusts for use in aircraft flight simulation. The approach combines principal components analysis, time series analysis and probability distribution model to analyze and simulate wind gust components. Comparisons are given between wind gust components generated by the model and components measured onboard an aircraft.

Nomenclature

a_i	Autoregressive coefficient
AR(p)	Autoregressive model of order p
ARIMA (p,d,q)	Autoregressive integrated moving average model of orders p and q with d differences
b_j	Moving average coefficient
e(t)	Random error
GLD	Generalized lambda distribution
MA(q)	Moving average model of order q
PCA	Principal components analysis
t	Time, sec.
UG, VG, WG	Three components of wind gust, m/sec
Z(t)	Time series variable
ZB	Mean of time series

Introduction

A realistic simulation of aircraft flight requires that all the control inputs and external forces be accurately modeled. One of the important external forces which must be modeled is the effect of wind gusts. Because of the random nature of wind gusts, realistic models of the horizontal and vertical gust components are difficult to obtain.

In this paper the random nature of wind gusts is tackled directly with a combination of three statistical tools. The three methods are applied in sequence both to analyze the gust components and to produce a model of the components. The three methods have been available in the literature for a number of years, but both the application and the combination of these methods are novel. After a model of wind gusts has been produced with these methods, the model can be used to generate gust components useful for input to aircraft flight simulations.

Method and Approach

In this section the three statistical techniques applied to wind gust data are described. The techniques are presented in the order in which they are applied to the data.

It is assumed the wind gust measurements consist of the three components of wind in a three-dimensional Cartesian coordinate system (UG,VG,WG), where UG and VG are horizontal axes and WG is vertical. In general the axes of this system will not be aligned with the wind direction. Because of this misalignment, any two of the three measured components will be correlated. If the components were not correlated, then each component could be analyzed separately.

The first task is, therefore, to transform to an axis system in which the measured components are uncorrelated. This is accomplished using principal components analysis (PCA). With PCA[1], the 3 by 3 covariance matrix of all the measurements is first calculated. Then the three eigenvalues and corresponding eigenvectors of the covariance matrix are calculated. The eigenvectors are used to form a 3 by 3 matrix which transforms the 3-component gust vectors to a new axis system called the principal components axes. In the new axes, the axis corresponding to the largest eigenvalue and associated eigenvector is called the first principal component. The largest eigenvalue is the variance of the transformed measurement component along the first principal component axis. The second and third principal components are similarly defined by the eigenvectors associated with the second and third largest eigenvalues and the eigenvalues are the variances of the transformed measurement components along these two axes. Essentially PCA transforms to a new set of orthogonal axes aligned with the major dispersion of the measurements; in doing this, the new components become uncorrelated.

Given a set of gust measurements consisting of 3-vectors whose components are uncorrelated, it is assumed each of the three sets of (transformed) components defines a time series (set of measurements across time). Therefore each component series can be analyzed separately using standard time series analysis techniques. The model chosen to represent each component series is an autoregressive integrated moving average[2] (ARIMA) model of orders p,d,q:

$$Z(t) = ZB + \sum_{i=1}^{p+d} a_i \; Z(t-i) + \sum_{j=1}^{q} b_j \; e(t-j) + e(t) \qquad (1)$$

In this equation, Z(t) corresponds to one of the three transformed gust components (U, V or W) and ZB is the mean value. The parameter d indicates the number of the times the original series was differenced; summation, the inverse of differencing, constitutes the "integrated" part of the model. The autoregressive model of order p (AR(p)) is given by the first summation which indicates the series is a function of past values. The moving average of order q (MA(q)) is given by the second summation and third term, where e(t) is a random input.

*Aero-Space Technologist.

The random error, e(t), and constant coefficients a_i and b_j are unknowns which must be determined. The approach taken here is to estimate the coefficients and the statistics of the random error using standard Box-Jenkins techniques[2] for an ARIMA (p,d,q) model. With this approach the user has the freedom to choose the number of differences (d) and the orders (p,q) of the AR and MA models which he considers most appropriate. The number of differences is chosen to remove linear and higher order trends; p and q are chosen to be small integers but large enough to provide a reasonable fit.

The Box-Jenkins approach assumes that the random error is normally distributed with zero mean and constant variance. Normally distributed random values also have zero skewness and a kurtosis (peakedness) value of 3. One of the objectives of this study is to determine if a non-normal distribution better represents the random error for use in simulating the wind gust. To accomplish this, the first four statistical moments (mean, variance, skewness and kurtosis) of the random error (residuals) found by the Box-Jenkins analysis are calculated. Then the generalized lambda distribution (GLD) of Ramberg and Schmeiser[3] is fit to the moments. The GLD is an analytical function which represents the inverse cumulative distribution function of a probability distribution. Distributions covering a wide range of skewness and kurtosis values, including that of the normal distribution, can be represented by proper selection of the GLD parameters. Given a GLD fit to the moments of the random error, the GLD can be used to generate values of the random error for input to the ARIMA model in order to simulate the wind gusts.

Simulated wind gusts are generated by reversing the operations of the three statistical analysis steps. First, a sequence of random error values are generated using either the GLD or a normal random number generator. These values are combined with the AR and MA parameters in the ARIMA model (eq. (1)) to generate a sequence of gust components in the principal components coordinates. Because three gust components are needed, three distinct sets of random error values are input to three different ARIMA models. The three sets of PCA gust components are combined to form a sequence of three-dimensional gust vectors over time. These vectors are then transformed back to the original gust coordinate system by applying the inverse of the 3 by 3 PCA transformation matrix.

Results and Discussion

The data used in this study was gathered during flights of a NASA F-106B into severe storm centers. This data consists of 1000 measurements of three Cartesian components of the wind gust velocities adjusted for the aircraft motion; the first two components (UG,VG) are horizontal components, the third (WG) is the vertical component. Several sets of measurements have been analyzed, but the results of only one set will be presented.

Figure 1 is a plot of the gust components over time; the horizontal components consist of very noisy trends. The vertical component contains an increasing linear trend, indicating strengthening updrafts encountered by the aircraft as it approaches the storm center. Figures 2 and 3 are plots of the autocorrelations and cross-correlations of the gust components. The slowly decreasing autocorrelations confirm the trends (i.e., nonstationary nature) of the components. All the cross-correlations, such as the cross-correlation of UG and VG, are non-zero, suggesting misalignment between the axes and wind direction.

The PCA transformation was calculated from the covariance matrix of this data. The results of transforming the data to the principle components axes are shown in Fig. 4. It can be seen the first principal component (UG) contains the greatest variation and the third (WG) contains the least. Furthermore, Fig. 5 shows all the cross-correlations have been reduced; in particular, the cross-correlations at zero lag are zero as prescribed by the PCA.

The Box-Jenkins algorithms used in this study are found in the IMSL software library.[4] Application of this software indicated differencing the data once was sufficient to remove major trends. Box and Jenkins[2] also gives criteria (pages 34, 65) for determining the AR and MA orders of the model. For the three principal components of the wind gusts these criteria indicated that p should be between 1 and 5 and q between 0 and 4. However in practice either the MA algorithm did not converge for q > 2 or the time series generated from higher order AR models was too large in magnitude.

Experimentation with values of p and q indicated the ARIMA (2,1,1) model gave an adequate representation of all three components. Analysis of the resulting residuals indicated the residual distribution had skewness between 0.2 and 1.0 and kurtosis between 5.3 and 6.7. These values suggest the GLD would be a better representation than would the normal distribution (skewness = 0, kurtosis = 3). The GLD parameters were estimated from the calculated moments with software developed at Langley Research Center.

Wind gusts were generated by inputting either GLD or normal random values into the ARIMA equations and transforming the ARIMA output with the PCA inverse transformation to the gust axes. Figures 6 and 7 present a comparison of the gusts generated with GLD and normal random values; except for slightly larger variation in the GLD-generated gusts, the two sets are very similar. The larger variation in GLD-generated gusts is evidently due to the positive skew in the distribution. Comparison of autocorrelations and cross-correlations of the two sets with those of the original gusts indicate the normal-generated gusts are more similar to the original gusts. Comparision of cross-correlations of figures 3 and 8 show significant differences in magnitudes of both the UW and VW cross-correlations. The differences are apparently due to inadequacies in the low order ARIMA model and the fact that the PCA transformation did not remove cross-correlations at all lag values. The latter condition conflicts with the assumption of independence made when fitting the ARIMA models.

Concluding Remarks

A method of modeling wind gusts which combines techniques from principal components analysis, time series analysis and probability distribution representation has been presented. The results of applying this method to gust measurements indicate the method may be useful for simulating wind gusts. The PCA transformation removed most of the cross-correlations between components in preparation for the Box-Jenkins time series analysis. The fitted ARIMA model provides a good representation of the components if appropriate AR and MA orders are chosen. However, based on the data sets examined, there appears to be no substantial advantage to representing the random error with the GLD. The major difficulty in this application is the nonstationarity of wind gust measurements, because the ARIMA model applies only to stationary data. The approach taken here to remove nonstationary effects by differencing the data appeared to be adequate. However removal of seasonal (periodic) effects was not attempted since available software for generating time series does not model these effects.

References

1. Anderson, T. W.: An Introduction to Multivariate Statistical Analysis. John Wiley and Sons; 1958.

2. Box, G. E. P.; and Jenkins, G. M.: Time Series Analysis, Revised ed., Holden-Day, 1976.

3. Ramberg, J. S.; and Schmeiser, B. W.: An Approximate Method for Generating Asymmetric Random Variables.Commun. ACM, Vol, 1, No. 2, Feb. 1974, pp. 78-82.

4. IMSL Library Reference Manual - Vol. 1, ed. 9. IMSL LIB-009 (Rev.), IMSL, Inc., 1982.

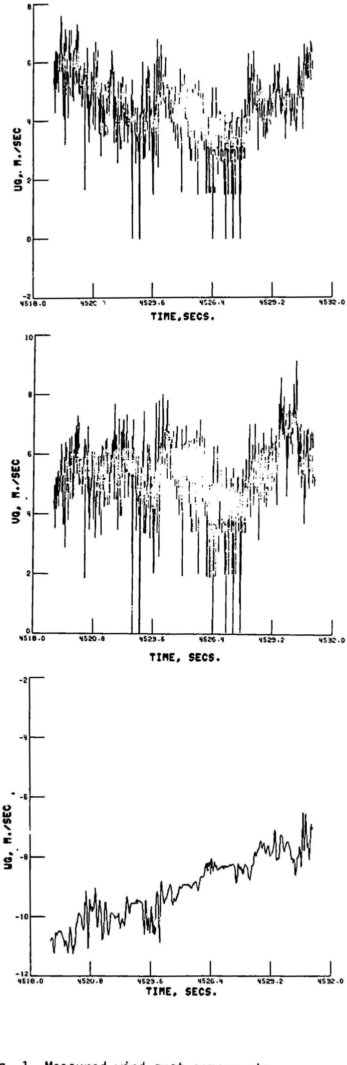


Fig. 1 Measured wind gust components.

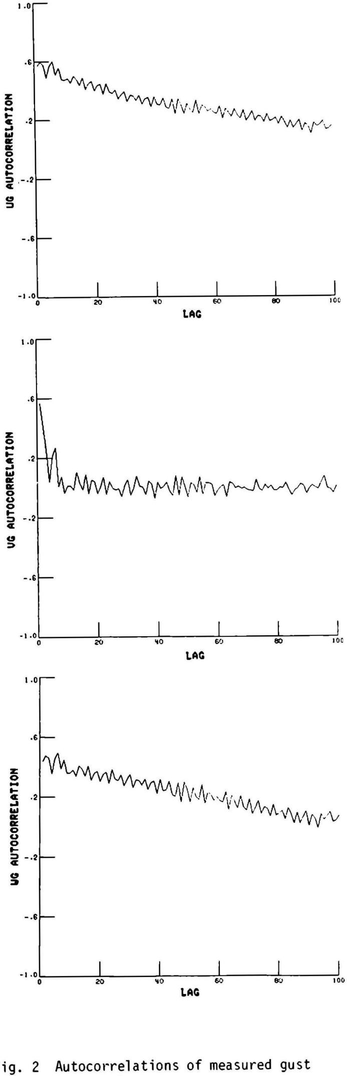


Fig. 2 Autocorrelations of measured gust components.

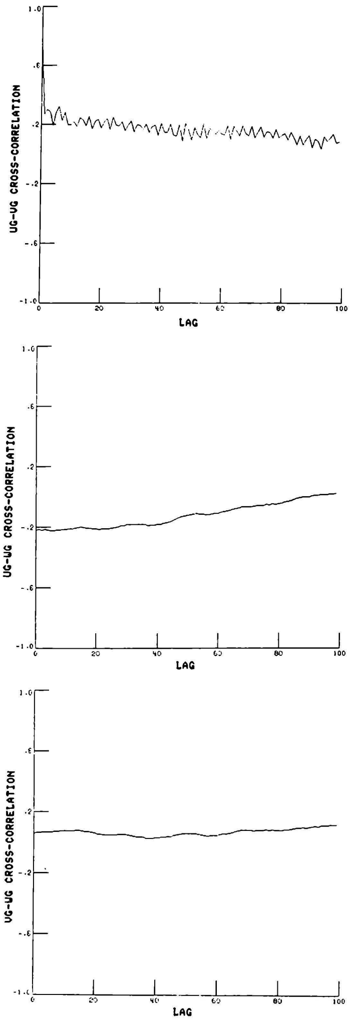


Fig. 3 Cross-correlations of measured gust components.

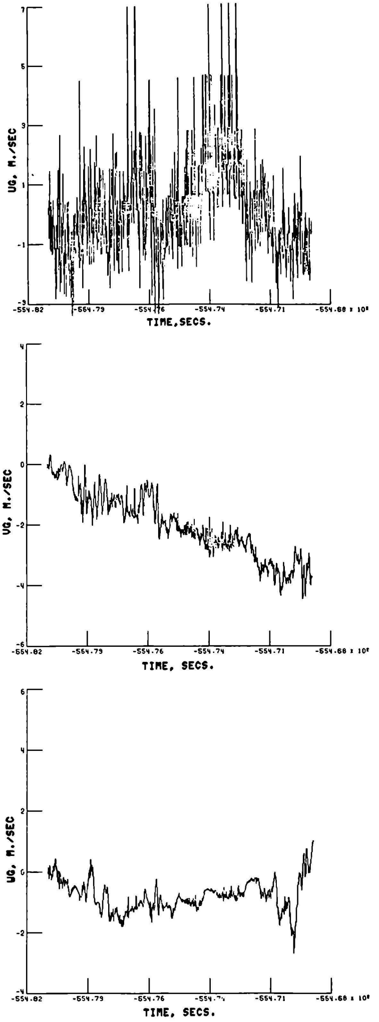


Fig. 4 Principal components of measured wind gusts.

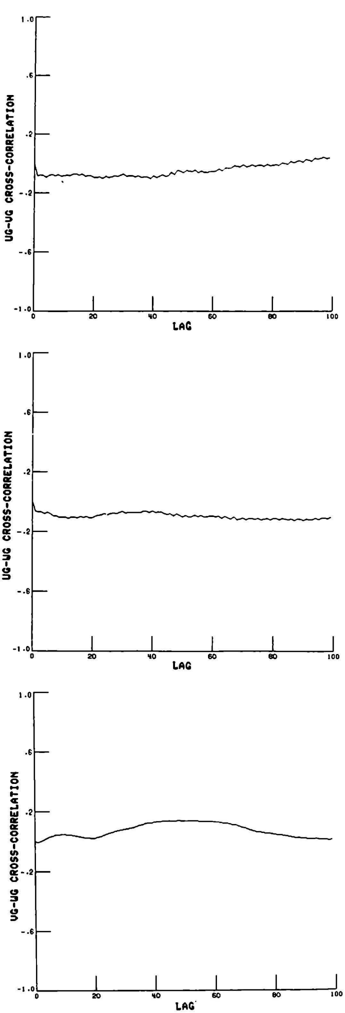


Fig. 5 Cross-correlations of principal components of measured wind gusts.

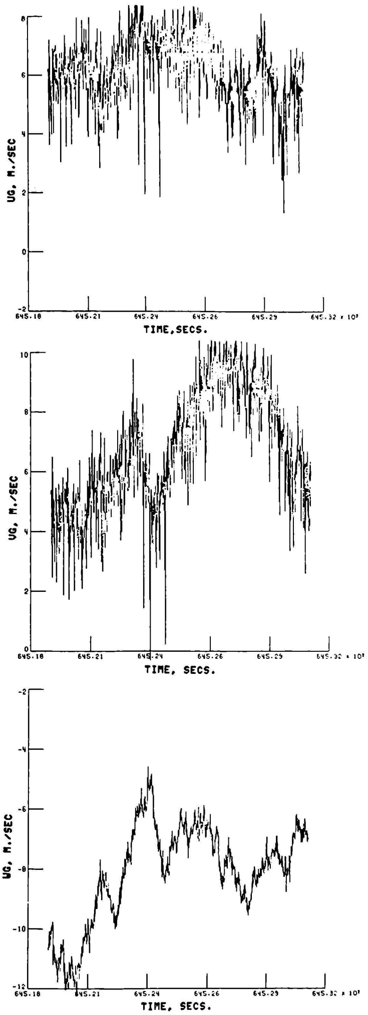


Fig. 6 Gust components simulated with GLD generated random input.

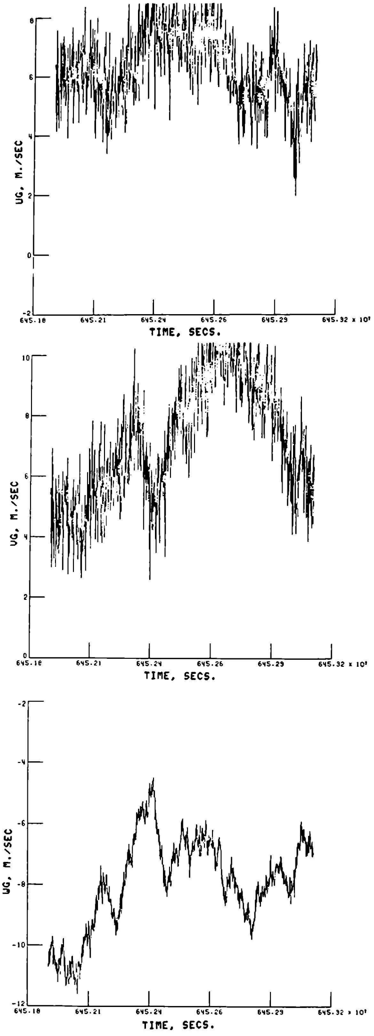


Fig. 7 Gust components simulated with Gaussian-generated random input.

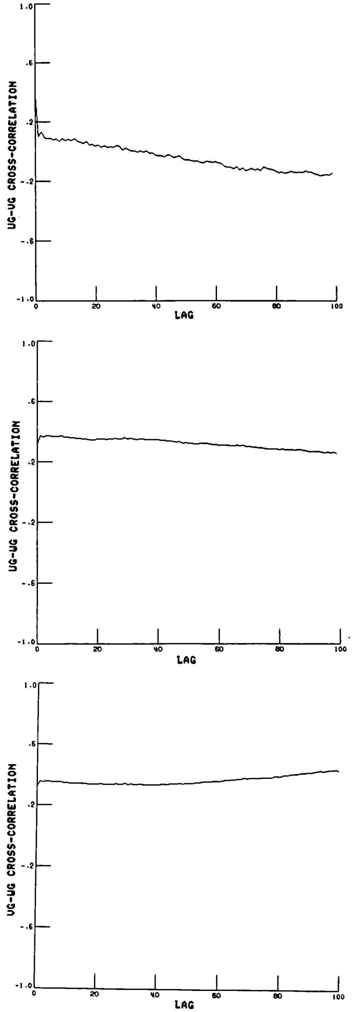


Fig. 8 Cross-corrections of Gaussian-generated wind gust components.

A FLEXIBLE DESIGN FOR A GENERAL PURPOSE SIMULATION FACILITY

Andrew C. Cruce, PhD
Scott A. Alexander

Systems Engineering Test Directorate
Naval Air Test Center
Patuxent River, Maryland 20670-5304

ABSTRACT

The Naval Air Test Center has been using simulation to support flight test for over eight years in the Tactical Avionics and Software Test and Evaluation Facility (TASTEF). Recently, this facility has undergone extensive redesign based on new requirements, new technology, and the experience gained with the original facility. This paper discusses the original and new facility architecture and components and discusses the reasons for the various changes and enhancements. The original facility was intended to support testing for a large variety of aircraft types and be capable of rapid reconfiguration to support multiple test programs simultaneously. Due to lessons learned and the application of new technology the new facility enhances these capabilities and greatly improves overall facility productivity.

BACKGROUND

Over the past seven to eight years, we at the Naval Air Test Center (NAVAIRTESTCEN), have gradually evolved what has become a uniquely flexible approach to avionics and flight systems simulation. In this paper we will trace the evolution of this system and the developing requirements that shaped its design. This continuing evolutionary process has allowed us the luxury of being able to live with some of our initial concepts over a period of time and then, to meet new requirements, modify the design of the initial system components to include lessons learned over a prolonged period of operation. The history of the design can be divided into two separate eras. The first was a relatively simple system that was used to support AYK-14, F/A-18, and AV-8B software testing. The second is our current expansion of this initial design into a true multiprocessor, multireal-time process simulation facility that is capable of simultaneously providing real-time simulation support to multiple independent projects.

THE INITIAL REQUIREMENTS: NAVAIRTESTCEN is the primary Test & Evaluation (T&E) site for all naval aircraft. During the T&E process we test all aspects of naval aircraft including weapons systems, flight systems, avionics systems, electronic warfare equipment, aircraft flying qualities, aircraft carrier suitability, and aircraft ground support systems. The type of aircraft tested covers everything from large transport type aircraft (P-3, C-130) to rotary wing aircraft (AH-1J, SH-60, JVX) and high performance fighter attack aircraft (F/A-18, AV-8B, F-14D, A-6F). In the mid-1970's we were faced with the dilemma of designing systems that would allow us to adequately test the new generation of modern fighter attack aircraft represented by the F/A-18 and the AV-8B. This problem is graphically presented in figure 1. The F/A-18 and other modern aircraft represent quantum leaps in the sophistication and complexity of aircraft weapons systems. By way of comparison the A-7E which represents the initial generation of digital system aircraft had a single digital computer with 16K words of memory. By the F/A-18, this onboard computer capability had grown to over 20 computers with several hundred thousand words of memory. Using the amount of software and firmware in the aircraft as a metric of the complexity of the system and thus a metric of the testing necessary to adequately characterize the system, we see that the test requirements for modern aircraft systems have grown immensely. At the same time the amount of flight test time available in a typical aircraft program to satisfy these test requirements has remained constant or in some cases decreased. The result is an ever increasing gap between the flight test time available and the flight test necessary to adequately T&E the system. As is common in the industry, we at NAVAIRTESTCEN turned to the use of simulation to leverage our available flight testing and allow us to make smarter and more efficient use of our flight test assets.

In developing a system to support our simulation requirements we analyzed our needs and attempted to identify the pertinent characteristics of a system that would satisfy these needs. The primary characteristic was that the simulation facility should provide for the reconfiguration and growth necessary to support the wide variety of aircraft types that are often under concurrent T&E at NAVAIRTESTCEN. The facility must also be capable of rapid configuration from one simulation to another to provide timely simulation support to the typically high tempo flight test operations conducted at NAVAIRTESTCEN. As you will see, the design of a system to meet these goals resulted in a system that was user friendly and provided a high degree of facility productivity.

THE INITIAL SYSTEM DESIGN: In its initial configuration TASTEF consisted of a number of PDP-11 computer systems connected as shown in figure 2. We will briefly describe the operatioı of this initial system since understanding how it operated will highlight the reasons for some of the changes we made in the second generation design. The intent of the simulation was to provide data to the Aircraft Mission Computers (MC's) over the 1553 buses that was identical in timing and content to the data they would see in the actual aircraft during flight. We wanted to "fool" the MC's in the facility into thinking they were flying in a real aircraft. If we were

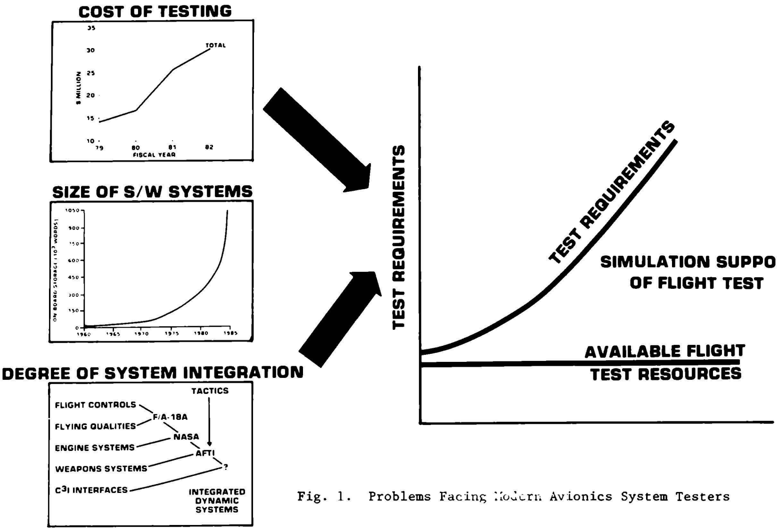


Fig. 1. Problems Facing Modern Avionics System Testers

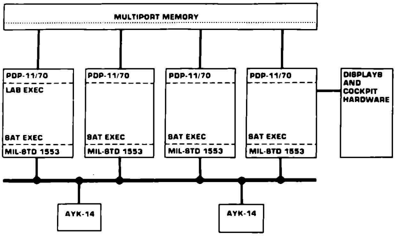


Fig. 2. Initial Facility Architecture

successful we could then T&E the Operational Flight Program (OFP) software in the facility on the ground and obtain meaningful estimates of the system performance.

The initial facility design to do this is shown in figure 2. The computers that made up the facility were connected via a common memory, an intercomputer interrupt line, and several MIL-STD-1553A buses. In order to simulate the F/A-18 system, two MC's were also connected to the buses which would run the actual F/A-18 OFP. The simulation process was simple. We wrote a series of environmental models including aircraft aerodynamics, oblate spheroid earth, atmosphere, and targets. We also wrote a series of avionics subsystem models that simulated all the avionics systems that communicated over the 1553 bus with the MC's. The aircraft aerodynamics model, combined with the targets model, provided the stimulus for the models of the aircraft avionics sensors. The environmental models calculated the state of the environment for each simulation frame and placed this data in the multiport memory. This memory was 12K bytes long and appeared to each machine to be logically appended to the end of each machine's local memory. It interfaced to each machine through the unibus and as a result required each machine in the facility to be within about 25 feet of the actual common memory hardware. The avionics sensor models peered into this memory in much the same way the actual sensors in the aircraft peer into the environment to sense current conditions. Once they understood the current conditions the sensor models could then properly respond to the requests for data from the MC's exactly as the real sensors in the aircraft. The other nonsensor models, such as the controls and displays and the Stores Management Set (SMS), did not use the environmental state information and were just designed to respond properly to commands for action or status from the MC.

The access to the data in the Multiport Memory was through the use of Shared Global Areas (SGA's) between the various models. To do this a data definition program was written that defined Fortran labeled commons that would reside in the SGA. Once this definition was complete and the data definition program properly built, it was used to define a SGA that resided in the Multiport

Memory. Models could gain access to the Multiport Memory by including the proper Fortran labeled commons in their subroutines and symbolically accessing the desired variables. This provided a great degree of flexibility since the linking process was automatic and both relieved the model developer from having to understand the intricacies of the simulation system and was independent of which particular machine on which the model was running.

The Digital Interfaces (DI's) that provided the PDP-11's with 1553 communication capability were also connected to the PDP-11 processors using the Unibus. These were microprogrammed devices using bipolar bit slice technology that implemented the complete MIL-STD-1553A protocol. The microcode, which was resident in PROM, was designed to allow any of the DI's to simultaneously act as up to eight 1553 Remote Terminals (RT's) or a single Bus Controller (BC). The BC mode was designed to emulate the BC mode of the AYK-14 where the BC IO is controlled by a chain program in AYK-14 memory. The operation of the RT mode of these devices is illustrated in figure 3. The interface showed up on the Unibus as eight sets of eight I/O registers corresponding to the eight separate RT's that the device could simulate. When two of the registers in a block of eight were loaded with a status word and an address table pointer for a particular RT the device was enabled to respond for that particular RT. Data coming over the bus was decoded and the address bits in the command word were used to determine if the command was for one of the active RT's in the device. If it was, then the address table pointer in the corresponding set of registers combined with the subaddress field and T/R bits in the command word were used to access the proper portion of the address table. The data in the address table pointed to the proper 32 word buffer that the data was either to be read from or written into. All the transfers between the DI and the PDP-11 were via DMA and controlled by the DI. Once the DI operation was initiated using a standard Digital Equipment Corporation RSX-11M QIO system call, the DI would continue to operate autonomously without further processor intervention.

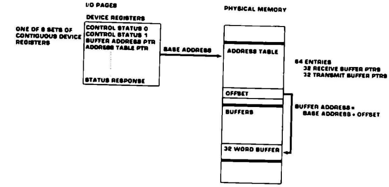


Fig. 3. Typical 1553A Interface Data Flow

In addition to the standard 1553 firmware, firmware was written to allow the DI's to function as smart bus monitor units and as special purpose 1553 interfaces that emulated the F/A-18 1553 display interface. In this second configuration, special firmware was written to allow the interface to properly interpret the 1553 bus commands to the F/A-18 display subsystems and construct a display file in PDP-11 memory that was bit-for-bit identical to the display file in the actual F/A-18 display hardware. The displays were then generated from these display files using special firmware written for an Adage 4145 display processor that emulated the F/A-18 display processor, and as a result properly interpreted the display files constructed by the OFP. A top level block diagram of this process is presented in figure 4.

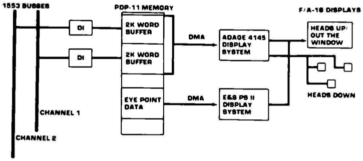


Fig. 4. F/A-18 Display Emulation

Using the system I have described so far, we were able to successfully simulate both the F/A-18 and AV-8B avionics systems. The block diagrams of the simulation architecture for each of these systems are shown in figures 5 and 6. A simple, single real-time process, multiprocessor, executive was designed to provide the initialization, control, and timing of these simulations. A configuration file like the one shown in Table 1 was used to drive the executive. This executive was designed during our initial efforts in bringing up the F/A-18 simulation and was used to distribute all the simulation models to the proper computers. Once the models were properly distributed, the executive also acted as a scheduler to cause each program to run properly during each simulation time frame, to check that the simulation was properly running in real-time, and to control starting, stopping, and single stepping of the simulation.

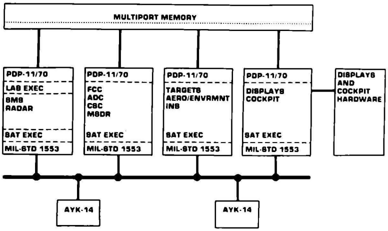


Fig. 5. Typical F/A-18 Simulation

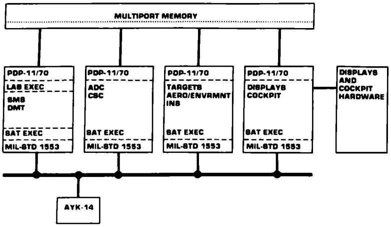


Fig. 6. Typical AV8B Simulation

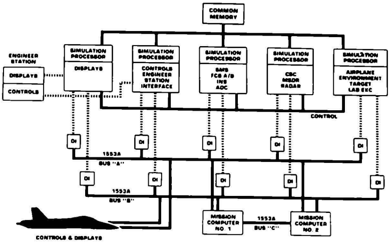


Fig. 7. Simulation with Actual Aircraft

```
-N42
[301,3]CKP.TSK/PRI:81.
[301,3]SWITCH.TSK/PRI:82.
-N71
[12,3]SRDROL.TSK/PRI:80.
[301,15]SMSCTL.TSK/PRI:80.
:[301,15]WRSTR.TSK
-N72
[301,11]INS.TSK/PRI:82.
[301,6]F18T.TSK/PRI:80.
-N74
[301,14]FCSMUX.TSK/PRI:83.
[301,2]ADC.TSK/PRI:82.
[301,10]CSC.TSK/PRI:81.
[301,13]MSDR.TSK/PRI:80.
-N73
:WORLD.TSK/PRI:80.
:[301,30]DDI.TSK/PRI:50.
[301,30]PICSW.TSK/PRI:50.
```

FIRST GENERATION CONFIGURATION FILE
TABLE 1

The AV-8B and F/A-18 simulations were used to support a number of T&E efforts during the course of these programs. They included air-to-ground weapon delivery accuracy and air-to-air launch acceptable region accuracy assessments. The system was also used to perform sensitivity analysis on the weapon delivery algorithms and to assist in planning for critical flight tests. It was also used to evaluate other hardware and software-in-the-loop in addition to the OFP-MC combination. These evaluations included prototype hardware such as a crash recorder system planned for the F/A-18 as well as a Research & Development test integration of a speech recognizer system into the F/A-18. They also included providing T&E support for other F/A-18 hardware. In particular the SMS system was evaluated using the simulation. The simulation configuration to do this was unusual and is shown in figure 7.

The SMS is the system in the aircraft that is in charge of weapons release and control. This includes providing the energy to the Cartridge Actuated Device (CAD) to release the weapon as well as providing conditioning information to the weapons such as missile moding commands, electrical fuzing signals, and control of mechanical fuzing options. In aircraft prior to the F/A-18, it was possible to T&E much of the operation of this system on the ground by hooking up recording instrumentation to the aircraft and then causing the SMS to go through its normal release sequences. In the F/A-18, however, the operation of the SMS is controlled by the OFP in the MC. The MC determines when to drop a weapon and commands the SMS to do this over the 1553 bus. The problem with the F/A-18 was that the MC and OFP were receiving information from other systems in the aircraft, such as the Inertial Navigation System (INS) and the Air Data Computer (ADC) indicating that the aircraft was not inflight and infact was stationary on the ground. Consequently, these systems would not perform the weapon release operations. As a result it was impossible, using the aircraft alone, to get the SMS to perform simulated normal weapons releases while on the ground. The hookup shown in figure 7 alleviated this problem. We connected the 1553 bus in the F/A-18 simulation directly to the 1553 bus in the aircraft. Once we had done this, the combination of the simulation facility and the actual F/A-18 aircraft hardware formed a complete system. We disabled all the systems in the F/A-18 with the exception of the Communications System Controller (CSC), the Controls and Displays, and the SMS. In the simulation we disabled the models of the CSC, the Controls and Displays, and the SMS. As a result, between the models in the simulation and the hardware in the aircraft, we had a complete F/A-18 system. We could now drive the environmental models to fly the aircraft through a simulated release and as a result stimulate the avionics system models to inform the MC that the aircraft was indeed on an actual weapon delivery run. Because of this, the MC would send the proper weapons release and moding commands to the SMS and the SMS would operate on the ground exactly as it operated inflight. This allowed engineers to connect instrumentation to the SMS and perform valid T&E of this system using ground testing.

INCREASED REQUIREMENTS: The system we have described to this point represented our initial operational capability to support T&E of aircraft avionics systems. As the usage of the system increased and as aircraft and computer technology continued to develop we were faced with increased requirements for the simulation facility. The primary drivers in the design of the upgraded simulation facility are listed:

Increased Utilization
More Aircraft Types
More Hardware-in-the-Loop Testing
SMS Testing Experience
Desire for Increased User Friendliness/High Productivity
Introduction of VAX Class Processor Technology
Introduction of MIL-STD-1553B

When broken down to its component elements our design for a flexible simulation facility is really composed of four separate elements. They are the executive that oversees and controls the simulation, the Multiport Memory that allows data sharing between simulation models, the DI's that allow the simulation to communicate with actual avionics system hardware and the simulation computers. As it turns out the drivers for change outlined above necessitated a redesign or enhancement of each of these component elements.

DEFINITION OF THE NEW SYSTEMS: The first five of the seven requirement drivers affect the executive and simulation computers. They result in a need for increased capability to control the reconfiguration of the facility and the ability to run multiple real-time simulations concurrently. The original executive allowed an operator to go into the simulation facility and by typing in a simple command, control the software configuration of the simulation computers. This was all that was necessary when we were only simulating a single aircraft type such as the F/A-18. In this case the other portions of the simulation such as the firmware loaded in the Adage display processor and the OFP code loaded in the MC's remained constant. However, as we moved to multiple aircraft simulations the engineer operating the facility was no longer able to type in a single simple command to the executive and get everything properly configured. Instead he was required to separately perform a number of hardware initialization functions such as load a new OFP, and/or new display microcode prior to initiation of simulation models.

Our experience with the SMS testing also indicated a need to redesign the executive to provide multiple, concurrent, real-time simulations. We had initially expected that the majority of our T&E data collection would be under software control. Under these kind of tests several hours of testing can often generate more data than an engineer can analyze in several weeks. As a result, our model of facility operation called for many short periods of intense use while either preparing for or actually conducting the computer controlled data collection process. In the SMS testing, avionics hardware was placed in the loop which prevented us from running the simulation at what amounted to be many times real-time. The result was that the SMS testing would take weeks of continuous real-time simulation during which the aircraft was attached to the simulation as shown in figure 7. This adversely impacted our operation because even though the simulation for the SMS T&E did not require anywhere near 100% of our computer simulation capacity it did take up 100% of our real-time executive capacity. In essence our continuing development was shutdown during SMS testing even though we still had substantial residual computer capacity. To cure this problem in the updated executive we decided to design in the capability to handle multiple real-time simulations concurrently.

Technology advances in general purpose computer systems since we originally configured our facility had resulted in a number of VAX-11/780 class computer systems. As part of our redesign effort it was necessary to determine the advantages of eventually replacing our PDP-11's with this newer technology. After evaluating several alternatives we selected the VAX-11/780. Strictly from the performance standpoint the VAX provided a greatly improved capability over the PDP-11's. We could in many cases run a complete simulation on a single VAX which had previously required four to five PDP-11's. The floating point format was compatible with the PDP-11's so the sharing of simulation variables between these machines was compatible. The major advantages to these machines however were related more to the software development environment. Our operational environment is very dynamic requiring the ability to respond rapidly to new requirements. This generally means the development and debug of lots of software. On the PDP-11's, debugging of Fortran models was frequently a tedious and time consuming process involving the iterative insertion of WRITE statements, recompilation, linking, and execution of programs. If the program was greater than 32K words this was further complicated by complex memory management considerations (i.e., in core overlays). On the VAX's, sophisticated debugging tools for high level languages dramatically improve the programmers productivity and addressing constraints disappear due to the large wordsize.

The last two requirement drivers dictated that we redesign the multiport memory and the DI's. The original Multiport Memory could accommodate up to eight PDP-11 processors and interfaced to these processors via the Unibus. As mentioned before this placed a fairly tight restriction on how far from the actual memory hardware the processors could be located. It also would have prevented us from including VAX-11/780 machines conveniently in the simulation mix. The reason is that although there is a Unibus interface on a VAX it will not support 32 bit operations to memory on that Unibus. As a result, we would not have been able to use the SGA approach to accessing variables from Fortran in a VAX system because a floating point Fortran reference in a VAX would have initiated a 32 bit floating point transfer. For this reason we decided to redesign the Multiport Memory to interface not only to the Unibus in PDP-11's, but also to the Synchronous Backplane Interconnect (SBI) in the VAX. We also decided to provide a host adaptor at each machine and then run our own ribbon cable from this host adaptor to the Multiport Memory. By doing this we could control the timing of the data transfers between the host adaptors and the Multiport Memory and as a result, ease the restriction on locating the Multiport Memory hardware immediately adjacent to the processors.

Finally there was the question of accommodating MIL-STD-1553B in the simulation. We could have provided this by a minor modification to the existing DI's together with a new set of microcode for the PROM's. However, we decided that improvements in the state-of-the-art of digital components were sufficient to warrant a redesign of the entire DI. This decision was supported by the desire to provide an increased flexibility in the redesigned DI's. The original DI's were programmable micromachines with the microcode stored in PROM's. As a result it was impossible to change the operation of these DI's without physically replacing the PROM set in the device. Since we wanted to be able to easily reconfigure a DI to act as a set of 1553A RT's, a set of 1553B RT's, a 1553A BC, a 1553B BC, a bus monitor, or a special purpose 1553 RT as in the F/A-18 display simulation, we decided that we should modify the DI design to incorporate a writable control store.

DESCRIPTION OF THE NEW SYSTEM: The architecture of the new simulation facility, shown in figure 8, is clearly a derivative of the original design. It still makes use of a Multiport Memory system to share data between processors in the simulation. Updated DI's still allow the simulation computers to communicate with the 1553 bus. The main differences are not in the basic architecture but rather in the capabilities of the components that make up the architecture. The Multiport Memory is bigger (up to 256K bytes) and can accommodate more computers (up to 16). These computers can be located up to 400 feet away from the memory hardware and can be an arbitrary mix of both PDP-11's and VAX-11/780 machines. Enhancements have also been added to the memory subsystem to provide system wide interrupt capability between any of the processors in the simulation and a system wide clock with an eight microsecond resolution.

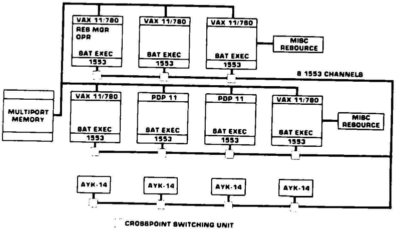


Fig. 8. Second Generation Architecture

The new DI's also provide a function similar to the old DI's only with a greatly enhanced capability. The new DI's are capable of simulating up to 32 1553 RT's simultaneously as opposed to the old eight unit capability. They also have a writable control store so that the same hardware can be used for a variety of functions such as bus controller simulations, 1553 simulations, 1553B simulations, and bus monitor functions by simply downloading new firmware. One new DI feature is the capability to interface with the facility "bus of 1553 buses". The two 1553 buses connecting the processors in the old architecture have been replaced with a bus of 16 1553 buses running throughout the facility. A dual redundant pair of these buses we call a channel. Each DI connects to a channel on this bus through connection hardware that can be thought of as a crosspoint switching network. The result is that under software control any DI in the system can be setup to connect with any of the eight channels in the facility. Similar crosspoint switching systems are also used to connect any actual avionics hardware to a particular one of the eight possible channels. The result is a flexible, reconfigurable system architecture that can accommodate virtually any current or proposed system architecture.

Finally, the executive has been extensively upgraded from a single process, multiprocessor, executive to a full multireal-time process, multiprocessor, executive. It runs under VMS and has been enhanced to perform facility resource allocation for each simulation. Rather than specifying the particular machine in the facility that a simulation model is to run on, a configuration file now only specifies the resources required by each model. A resource manager portion of the executive matches the resources required by each model with the resources available on each processor, including available memory and CPU bandwidth, and automatically allocates the model to a processor on which it can run. This allows the executive to juggle the models between processors when one simulation is already running and another one is queued up to run simultaneously.

DETAILED HARDWARE/SOFTWARE DESCRIPTIONS: In the next several pages we will present more complete descriptions of the hardware and software elements that constitute the facility architecture. Specifically we will address the:

MIL-STD-1553 Digital Interfaces (DI's)
Bus Switching Unit
Multiport Memory
Laboratory Executive

MIL-STD-1553 Digital Interface (DI) -- The DI is a user microprogrammable device that communicates between a DEC Unibus and a MIL-STD-1553 A or B multiplex bus. Figure 9 shows a top level block diagram of the interface. It has an internal,

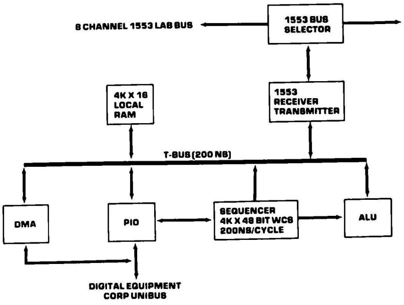

8 CHANNEL 1553 LAB BUS
1553 BUS SELECTOR
4K X 16 LOCAL RAM
1553 RECEIVER TRANSMITTER
T-BUS(200 NS)
DMA
PIO
SEQUENCER 4K X 48 BIT WCS 200NS/CYCLE
ALU
DIGITAL EQUIPMENT CORP UNIBUS

Fig. 9. Programmable MIL-STD 1553 Interface Block Diagram

synchronous, 200ns, bus with an AMD 2910 microprocessor and an AMD 29116 Arithmetic Logic Unit (ALU). The internal logic provides 4K words of user writable control store, 8K words of scratchpad memory and an interface to the 1553 bus. Two paths are provided for communication with the DEC host. During normal 1553 I/O there is a Direct Memory Access (DMA) path directly to memory through the Unibus. For initiation, control and microcode debug there is a Programmed I/O (PIO) path from the host processor to the interface. Using the PIO path, software in the DEC host can access any of the devices on the internal interface bus, load and inspect internal scratchpad memory and registers, and control operation of the microprocessor. Special firmware debug features have been included such as a breakpoint micro-instruction and a built-in logic analyzer on the micro-address bus for micro-program tracing. These capabilities greatly enhance the programmers ability to produce special purpose firmware. The microprogrammable character of the interface makes its use extremely general. It can be programmed to be a set of normal MIL-STD-1553 A or B remote terminals and bus controller, a special purpose MIL-STD-1553 interface such as the F/A-18 or AV-8B display system, a bus monitor system, or any other type of device. For purposes of illustration we will describe how the device is used as a normal MIL-STD-1553 remote terminal interface. Initially, support software under control of the laboratory executive loads the interface with the proper microcode using the PIO interface. Once the microcode is loaded, the models that will communicate with the interface are initiated by the laboratory executive and setup links to the interface using a custom I/O driver. During operation the interface continually monitors the bus and, on receipt of a message directed to any of the remote terminals that have been initiated in the interface, takes the appropriate actions. This includes storing incoming data in data buffers in the models or using data buffers in the models to respond to requests for information from the bus controller. The power and flexibility of the device is illustrated by the fact that the total microcode necessary to simulate up to 32 remote terminals on a MIL-STD-1553A bus only requires about 256 words of microcode out of 4K words available.

Bus Switching Unit -- The bus switching unit is used by the laboratory executive to set up the proper bus configuration for a simulation. Since the facility is designed to be multipurpose and reconfigurable it is necessary to provide some simple method of reconfiguring the facility bus structure. The Bus Switching Unit supplies this capability. As figure 8 shows the MIL-STD-1553 interfaces and actual avionics hardware in the facility are connected to a bus of eight MIL-STD-1553 buses through the bus switching unit. The bus switching unit is a crosspoint switch that can connect each interface or piece of avionics equipment in the facility to one of the eight possible buses under computer control. These switches are controlled by a laboratory executive that uses a configuration definition file to determine what equipment must be on a particular bus for a particular simulation. During the simulation initialization phase the laboratory executive uses the bus switching units to setup the required bus architecture for a particular simulation.

Multiport Memory -- The Multiport Memory is the hardware that allows the models in the simulation to share data. This system provides up to 256K bytes of memory that can be shared between up to 16 processors in the facility. It is designed to interface with PDP-11 machines through the Unibus and with VAX-11/780 machines through the SBI. The memory in the Multiport Memory System appears to be logically present in all machines connected to the Multiport Memory simultaneously. During initialization the laboratory executive assigns a segment of the Multiport memory to a particular simulation. An SGA is installed in this portion of the memory and all the models in a particular simulation use this SGA to symbolically share data. The Multiport Memory has two features that have been specifically designed with simulation requirements in mind. The first allows a wide range of distances between processors connected to the memory. By changing the timing on the interface between a particular processor and the Multiport Memory, it is possible to have up to a 400 foot separation between any processor and the Multiport Memory hardware. The second and more important feature provides a hardware solution to the normal simulation problem of mixing data from two time frames in a simulation. The general sequence of updating a simulation model as a function of time is to take data that are valid for some time t and use it to calculate the new state of the simulation at the next time frame t+dt. A problem that arises in many simulation facilities is that once any t+dt values are calculated they must somehow be segregated from the time t values to avoid mixing the two time values in subsequent calculations. The Multiport Memory provides a hardware solution to this problem. Portions of the memory can be setup so that the data are double buffered. There is a "write only" section and a "read only" section. A read to the same virtual location will return the value valid at time t which was written into the "write only" section on the previous time frame. At the end of each iteration interval, the laboratory executive switches the "read only" and "write only" buffers so that at time t+dt the data calculated to be valid at time t+dt is in the "read only" section.

The Multiport Memory system also provides the capability to selectively interrupt any or all of the processors in the facility. There are four registers provided in the Multiport Memory system that correspond to four interrupts that can be generated in any of the machines connected to the system. These interrupts are generated by writing a bit mask into one of the registers with the bits set corresponding to the machines to be interrupted. If a machine to be interrupted has not disabled the particular interrupt issued then it is interrupted. This feature is used by the laboratory executive to schedule and coordinate the operation of the processors in the facility. One additional feature added to the multiport memory was an eight µs common clock register. This register is a 32 bit counter which may be cleared periodically and read as desired to provide timing synchronization between simulation models on the same or different processors. With an eight µ sec resolution overflow will not occur for over nine hours of continuous simulation.

Laboratory Executive -- The Laboratory Executive is the central software system that controls the execution and timing of simulation programs running in the facility. It is a multiprocessor, multiprocess executive that is capable of controlling multiple, concurrent, real-time simulations. The executive is designed to provide an unsophisticated user simple yet complete access to all the capabilities of the TASTEF. A design goal was for a user to be able to walk into the facility, type a few characters on a terminal and have the entire facility configured to his particular simulation within seconds. This capability has been provided by designing the executive to work with a configuration file that defines a complete simulation environment. Table 2 gives an example of this file. It contains all the information necessary for the executive to configure the system. It tells the executive what hardware resources are required in the simulation (mission computers, 1553 buses, 1553 interfaces, etc.), the configuration of these resources (microcode load for the interfaces, OFP for the mission computers), and the simulation software to be run. The executive interprets this file, checks to see what resouces are available in the laboratory and if the proper resources are available configures the facility to run the simulation. In specifying the software to be run in the simulation, the user has the option of specifying the particular computer in which the software is to be run or letting the executive place the software in a particular computer. The philosophy behind letting the executive choose the computer is based on the multiple, concurrent real-time simulation capability of the system. Many of the models that run in a simulation in the facility architecture can be run in any of a number of computers in the facility. The only resources they require are access to the Multiport Memory and access to a DI. When multiple simulations are running concurrently, the executive keeps track of the CPU loading of each of the computers in the facility and uses this information to choose a particular computer to place a model from a new simulation it is initiating. The executive also keeps track of the loading in each of the computers to determine if there are sufficient resources and CPU bandwidth left in the facility to start another real-time simulation. Tables are used to provide the executive with information on the hardware configuration of the facility. This means that as new elements are added to the facility or if a particular resource in the facility is for some reason not available, the executive can be modified to reflect this by simply changing an entry in a table.

FACILITY OPERATION: All the elements described previously are tied together using the laboratory executive to produce a comprehensive simulation capability. A particular simulation is defined by specifying the hardware and software configuration that is to run in the laboratory. The laboratory executive allows the user to specify this configuration in a configuration file and then easily return to this configuration each time the simulation is run. The simplest way to completely illustrate the operation of the system is to go through a detailed example of setting up a particular simulation and then running it. For this illustration assume we desire to simulate the simplified F/A-18 system shown in Figure 10. This system has two AYK-14 mission computers, a full set of F/A-18 displays and a limited set of avionics subsystems consisting of the flight control system, the radar, the CSC, the SMS, the ADC, and the INS. We will also assume that the simulation configuration is being set up by an engineer who will be making modifications to the radar model. As you will see, the laboratory executive will allow the engineer to set up the simulation configuration to use baseline models for all the systems except the radar and at the same time use the latest development configuration of the radar model.

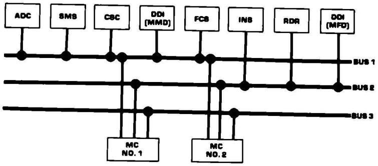


Fig. 10. Simplified F/A-18 Simulation Example

The first step in setting up the simulation configuration is to define exactly what this configuration is to be. Figure 11 shows an approach for doing this. The various models that comprise the simulation are divided according to the particular resources they require. Four of the models, the ADC, SMS, CSC, and FCS must be connected to bus 1 in the aircraft. They are shown in one block in the simulation definition connected to this bus through a DI that has been loaded with microcode to simulate MIL-STD-1553A remote terminals. Similarly, the INS and RDR models must be connected to bus 2 through a DI running MIL-STD-1553A remote terminal firmware. In order to assist in debugging the radar model, the engineer also wishes to put a bus monitor device on the same bus as the radar model. The display models must be connected to both buses 1 and 2. This is done through two separate interfaces which are loaded with firmware to emulate the DDI interface to the MIL-STD-1553 bus. It is assumed that the engineer wishes to use cockpit 0 which includes both a Picture System (PSO) for the out-the-window view and an Adage 4145 (ADO) to decode the DDI display file. The airframe model to be used is F18AIR and the cockpit interface task is F18CKP. Finally the OFP requires two AYK-14 computers, in the XN-5A configuration, that are connected to three buses as shown in figure 11. These two computers are loaded with OFP's 029 and 030 respectively.

Table 2 contains the configuration file that describes this simulation configuration. The first line of this file is a job order number that is used for accounting purposes. The second line is the primary cycle time of the simulation in milliseconds. In this case the simulation is to run 20 times per second. The next line is used to tell the laboratory executive the number of 4K blocks of multiport the simulation requires. The last line in this initial portion of the file indicates several of the resources that will be

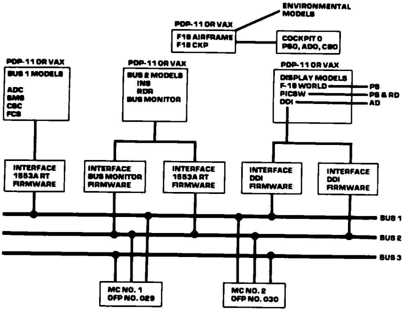


Fig. 11. Simulation Configuration

```
; GLOBAL SIMULATION PARAMETERS
/JO = F181691SY ; JOB ORDER NUMBER
/CYCLE=50        ; SIMULATION CYCLE PERIOD (MS)
/BL=1         ; COMMON MEMORY REQUIREMENTS (4 KW
BLOCKS)
/RES=(CPO.AYA0.AYA1); COCKPIT 0 RESOURCES REQUIRED
;
;  DEFINE 1553 BUS UTILIZATION
;   DEFINES CONNECTION OF LOGICAL INTERFACE TO
LOGICAL CHANNEL
;
/DEV:IN0=CH0      ; NORMAL RT'S CHANNEL 0
/DEV:IN1=CH1      ; NORMAL RT'S CHANNEL 1
/DEV:IN2=CH0      ; DISPLAY RT CHANNEL 0
/DEV:IN3=CH1      ; DISPLAY RT CHANNEL 1
/DEV:IN4=CH1      ; SPECIAL BUS MONITOR CHANNEL 1
#
; -SPECIAL FUNCTION PROCESSORS-
;
; THESE SPECIAL PROGRAMS EXECUTE DURING SIMULATION
INITIATION
; TO LOAD SPECIAL DEVICE FIRMWARE OR PERFORM
DEVICE DEPENDENT
; INITIALIZATION
;
MICLDBUS/RES=IN0/PARM="RT1553A"
MICLDBUS/RES=IN1/PARM="RT1553A"
MICLDBUS/RES=IN2/PARM="F18DISPLY"
MICLDBUS/RES=IN3/PARM="F18DISPLY"
MICLDBUS/RES=IN4/PARM="BUSMONITR"
AYKLOAD/RES=AYA0/BUS=(CH0,CH1,CH2)/PARM="MC029"
AYKLOAD/RES=AYA1/BUS=(CH0,CH1,CH2)/PARM="MC030"
ADGLOAD/RES=AD0/PARM="F18"
%
;  SYNCHRONOUS MODELS
;
; THESE MODELS ARE SYNCHRONIZED BY THE RESOURCE
; MANAGER
;
[BRUCE]F18AIRFRAME.EXE/PRI=18/TIME=20
[BRUCE]AIRINIT1.DAT/NAME=AIRINIT.DAT
F18FCS.EXE/PRI=19/TIME=4/NODE=N82/RES=IN0
N72=DB0:[DAVE]F18CKP.EXE/TI=1/RES=CB0(81)
N71=F18SWTCH.EXE/TIME=1/RES=CB0(81)
?N81=[GENE]RADAR.TSK/TIME=17/RES=IN1/NODE=N82
N71=CSC.TSK/TI=6/RES=IN0
N71=SMS.TSK/TIME=28/NODE=N74/RES=IN0
ADC.TSK/TIME=5/RES=IN0
INS.EXE/TIME=2/RES=IN1
:
;
; ASYNCHRONOUS MODELS
;
N71=DB1:[301,15]F18WORLD/RES=PS0/TI=20
N71=DB1:[301,15]PICSW/RES=(PS0.AD0)/TI=1
N71=DB1:[301,15]DDI/TIME=1/RES=(AD0.IN2.IN3)
N71=[301,3]MONITOR/RES=IN4/DEBUG=TTA4
```

SECOND GENERATION CONFIGURATION FILE
TABLE 2

required by the simulation. In this case cockpit 0 is specified as required along with two AYK-14 assets in the XN-5A configuration. The cockpit 0 specification, CP0, is a macro definition. This means that in the resource definition file, cockpit 0 is defined to include the Picture System (PS) II and Adage (AD) graphics system associated with cockpit 0 as well as cockpit bus 0 (CB0). The syntax of the configuration file uses keywords and requires no special sequence in specifying particular items. As the example shows, after encountering a semicolon in a line the laboratory executive treats the remainder of the line as a comment. This allows for extensive documentation within the configuration file.

The next series of definitions in the configuration file allocate logical interface specifications to logical buses in the laboratory. In this case interface types IN0 and IN2 are defined as being on logical channel 0 in the laboratory. Likewise, interface types IN1, IN3, and IN4 are defined as being on logical channel 1. The reason for specifying different interface types on the same channel is that these interface types will be loaded with different microcode to perform desired functions. The microcode load for each type of interface is defined in the next block of commands to the executive. The filename MICLDBUS identifies a program called a special function processor that is used to load microcode into DI's. The figure shows that interface types IN0 and IN1 are to be loaded with MIL-STD-1553A microcode, types IN2 and IN3 are to be loaded with display related microcode, and IN4 is to be loaded with bus monitor microcode. The same kinds of processes are used to load the AYK-14's with a specific OFP (OFP's 029 and 030) and to load the Adage graphics system with microcode that emulates the F18 display processor.

Some discussion is necessary concerning the relation between interface types, logical channel numbers, and physical bus channels. In the TASTEF there is a "bus" of eight MIL-STD-1553 channels that are connected through bus switching units to the devices in the facility that can connect to MIL-STD-1553. These devices include the MIL-STD-1553 interface units as well as any actual avionics equipment. In processing the configuration file the laboratory executive determines how many logical channels are required in the simulation. In the example presented in table 2, the configuration file specifies three of these logical channels (CH0, CH1, and CH2). It then determines which of the eight physical channels in the laboratory are not in use and allocates a separate physical channel to each logical channel. Each interface type is also specified to connect to a particular logical bus. This tells the executive how to configure the bus switching unit on each interface in the simulation to connect it to the proper physical bus. The interface types (IN0, IN1, etc.) do not necessarily refer to a particular physical interface. In specifying these types the executive is told to which logical bus they connect and what microcode they are to run. One or more programs to be run in the simulation may specify the requirement to connect to a particular interface type. However, if the executive places different programs that require a given interface type on different machines then there may be one or more physical interfaces that are defined under a single interface type definition.

The contents of the configuration file up to the "%" define the bus layout and special requirements of a given simulation. All that remains to complete the simulation definition is to specify what simulation programs are to run: The first set of program specifications, those between the "%" and the ":", are tasks that are to be synchronized by the executive. A typical program specification includes the name of the program, the priority at which it is to run, special resource requirements, and the worst case execution time in milliseconds. Several examples will illustrate the syntax of the configuration. In the radar model specification in table 2 the "?" at the beginning of the line causes the laboratory executive to ask the engineer, each time this simulation configuration is run, if a new version of the radar model is desired. If the answer is yes, then the executive obtains the file specified in the configuration file and moves it to the development directory on the target machine. In this case the file [GENE]RADAR.EXE would be moved from node 81 where the engineer is developing the software to the development directory on node 82 where the model is specified to run.

This example illustrates some of the configuration management capabilities that are designed into the laboratory executive. There are two distinct phases in the life of a model in the facility. The first phase is during development when the configuration and operation of the model are under rapid and continuous change. The second is when a model has been completed, debugged, tested, and validated. The laboratory executive provides a structure to keep software in each of these phases of life separate. In the example in table 2, all the models, with the exception of the radar model, are obtained from a baseline directory. Models stored in this area are assumed to be complete and working. The fact that the engineer has told the executive to ask if a new copy of the radar model is desired each time the simulation is run, identifies the radar model as developmental in nature. As a result the executive goes to a different directory, the development directory, to find the radar model if a new version is not desired. If the engineer answers the executive query to ship a new version affirmatively, then a new copy of the file [GENE]RADAR.EXE is moved from node 81 to the development directory on node 82. This structure allows various levels of configuration management formality in an organization using the laboratory executive. If highly structured configuration management is desired then a single configuration manager can be assigned to examine each model and verify it is complete and operating properly before it is transferred from the development to the baseline directory. For organizations with less formal configuration management requirements the process of moving a model from a development to a baseline directory can be correspondingly eased.

Continuing with our example, the remainder of the radar model specification provides the executive with the worst case time required for the radar model to execute in milliseconds (TIME=17), the interface type and as a result physical bus that the model is to connect to (RES=IN2), and the particular processor that the model is to run on (NODE=N82). The remainder of the synchronous models are specified in a similar manner. One point to note is that not all of the model specifications include the node on which the model is to be run. In fact most of the model specifications do not include this information. The reason for this is to allow the executive to allocate models to any processor in the facility that has the resources required by the model. This flexibility is what enables the executive to run multiple, concurrent, real-time, simulations. When one simulation is already running in the facility and the executive is requested to run a second simulation, it uses the resource requirements for the new simulation to determine if there are sufficient resources remaining in the facility to run the second simulation. If a model does not have to be run in a specific machine then the executive has a great deal more flexibility in allocating models within the facility and being able to take maximum advantage of the total resources in the facility.

The first set of programs, those between the "%" and the ":", are synchronized by the executive. This means that each model is run individually at the cycle rate specified at the beginning of the configuration file. Models use Fortran library routines to connect to the executive. Figure 12 shows the general structure of models designed to be run in the TASTEF. At the very beginning of the model a call is made to SINIT. This call informs the executive that the model is capable of running and takes care of some bookkeeping operations between the model and the executive. Next the model performs whatever initialization functions are required and when they are complete calls SYNC. This is the routine that is used to start and stop the model under control of the executive. When the executive determines that all the models have completed their initialization it frees them to run. Each model is designed to perform its cyclic processing and then before starting the next cycle to call SYNC. In this way each model is run individually during the simulation iteration cycle. The executive also keeps track of the execution time of each of the models and if for any reason some of the models do not complete in the specified iteration cycle it will inform the engineer running the simulation that the simulation has dropped out of real-time.

The second set of programs specified in the configuration file are asynchronous models. These are models that, for a variety of reasons are required to execute at their own rate rather than be tied to the basic iteration rate of the simulation. In some cases, such as display models, this allows the models to execute at a higher rate than the total simulation. In other cases, where real-time execution is not required

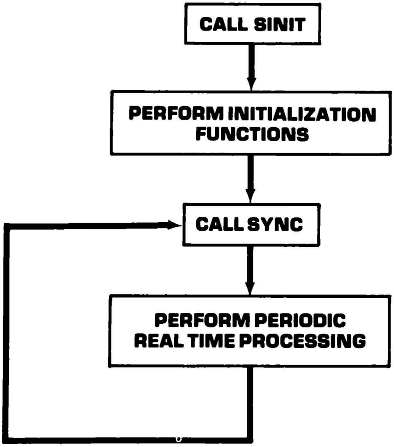

CALL SINIT
PERFORM INITIALIZATION FUNCTIONS
CALL SYNC
PERFORM PERIODIC REAL TIME PROCESSING

Fig. 12. Structure of Simulation Tasks under the Laboratory Executive

of a specific model or program, the model or program iteration rate may be much slower than the simulation. With these exceptions the syntax of specifying these models is identical to the synchronous models.

The operation of the simulation facility using the laboratory executive is extremely simple. All an engineer has to do to start a simulation is to enter a command OPR FILENAME where the filename is the name of the configuration file that defines the desired simulation. At this point the executive takes care of properly configuring the facility, loading the proper software in the mission computers, setting up the proper DI's, and connecting them to the proper buses, and running all the required models. By typing this single command to the executive an engineer can reconfigure the facility from a simulation of one aircraft type to a completely different aircraft type within 30-90 seconds.

One final point to cover concerning the TASTEF architecture is its extendability. The number of processors that can be accommodated in the facility is limited by the number of processors that can share the multiport memory. This is currently limited by hardware to 16 processors. The executive is table driven using files that define the current configuration of the facility. This means that as requirements for facility usage increase additional processors can be easily accommodated in the TASTEF architecture by only changing the facility definition file that drives the executive. This allows extremely easy and rapid facility capability response to changing facility requirements.

FUTURE PLANS: Figure 13 indicates that the continued development of this simulation facility will not stop with the current second generation hardware and software. Instead the updated system described here will form the kernel of a full engineering flight simulation. This system, known as the Manned Flight Simulator (MFS), will include a high fidelity Computer Image Generation (CIG) system that will drive two simulation bays. The first will be a six- degree-of-freedom motion system with a limited field-of-view world display.

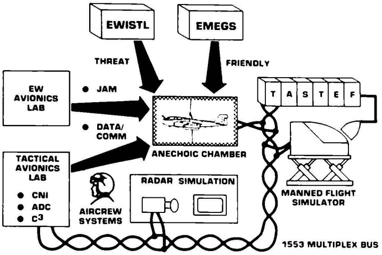


Fig. 13. Fully Integrated Facility

This will be used to support VSTOL, hover, and transition flight regime testing as well as non-centerline thrust flying qualities evaluations. The second will be a complete spherical field-of-view world display with high resolution images suitable for Air Combat Maneuvering (ACM) as well as nap of the earth air-to-ground missions.

The MFS in turn forms the kernel for further expansion to a full Air Combat Environment Test and Evaluation Facility (ACETEF). In the ACETEF concept existing assets, such as an aircraft sized anechoic chamber and extensive ESM and ECM stimulation hardware, will be coupled with the MFS to provide a realistic combat environment for both the pilot and the aircraft. When operating in its fully integrated mode, a pilot in the MFS will fly through a visual threat environment. The simulation state generated by the pilot inputs will be used to control a sophisticated ESM stimulation system that will provide realistic, high density, threat inputs to the ESM system of a fully integrated aircraft hanging in the anechoic chamber. At the same time the avionics simulation in the TASTEF will provide simulations of the avionics subsystems, such as the INS and ADC, necessary to fool the aircraft in the chamber into thinking it is inflight. In addition, sophisticated, closed-loop, threat missile simulations will be used to evaluate the performance of the ECM system.

In a typical scenario the pilot will fly the system against some realistic threat environment. The ESM system will be stressed by numerous realistic signals that include both friendly and hostile radars as well as other electromagnetic signals. While the pilot attempts to perform some task, such as delivering a weapon, the ESM system will have to sort out all the incoming electromagnetic signals to determine if any active threats are present. When an active threat is found the combination of the ESM system, the ECM system, the OFP, and any ARM hardware will have to take proper actions to neutralize the threat. This might include active jamming as well as providing commands to the pilot to take evasive action. Accurate simulations of the actual threat missile will be used to evaluate the effect of these neutralizing tactics and evaluate the performance of the overall system. The intent is, as the name implies, to obtain accurate estimates of the system performance and survivability in an actual combat situation.

85-1754

A Real Time Executive For The Control of a Multicomputer Simulation Complex

Jeffrey H. Croft
The Boeing Company
Seattle, Washington

ABSTRACT

Providing the capability to control and synchronize multiple computers in a real time environment requires a sophisticated simulation executive. Such an executive was developed for use in the Boeing B-1B Avionics real time simulation. The simulation complex consists of four Harris 800 computers connected by shared memory. These simulation computers communicate with six operational Avionic Control Units (ACU) via multiple MIL-STD-1553B buses. The simulation computers contain software models of the subsystems needed to interface with the flight software in a closed loop real time mode. The simulation complex is used to provide the interfaces needed by the operational flight software but not available in a laboratory environment. In order to provide the needed interfaces, the simulation software consists of a Vehicle System Simulation (VSS), Weapon System Simulation (WSS), Defensive System Simulation (DSS), and Radar Data Simulation (RDS). Each of these four simulations reside in a separate Harris 800 computer along with the simulation executive. The executive provides the interface between the simulation software and hardware. The simulation executive replaces the Harris supplied operating system and provides complete process and I/O control of the simulation software. This development of an entire operating system was necessary because of the lack of any commercially available or previously developed executive which could satisfy all or a major portion of the simulation operating requirements. The developed executive provides an excellent basis for future simulation complexes hosted on Harris computers.

INTRODUCTION

As avionic systems become larger and more sophisticated, the need for bigger and better simulation complexes are required. The heart of all avionic systems consists of a computer or computers which communicate with other line replacable units (LRUs) and subsystems on the air vehicle. When the software within the avionic computers is being developed, the other computers, LRUs, and subsystems are often not available because they are also being developed at the same time. In order to develop the flight software the interfacing LRUs and subsystems must be simulated in a real time, closed loop fashion. That is, the responses or commands from the simulated LRUs and subsystems must be the same as what the responses or commands would be under the same set of conditions with the real LRUs and subsystems. Any inherent delays or data availability restrictions present in the real LRUs and subsystems must be considered and handled in the corresponding simulations.

The ability to add a real LRU or subsystem to the simulation complex and turn off the corresponding simulated LRU or subsystem is required. An example is an interface unit to the weapons in the air vehicle. In one mode both the interface unit and weapons must be simulated. In the other mode the real interface unit is used and just the weapons need to be simulated.

In addition to LRU and subsystem simulations, the simulation complex must provide other features and capabilities. An environment for the simulation models to execute under must be provided as well as a human interface in order to control and monitor the models. Support functions must also be included which allow for the gathering and analysis of data and the faulting of simulated devices. All these additional features make a simulation complex a powerful tool. The simulation complex addressed in this paper supports The Boeing Company's B-1B Avionics development program.

SIMULATION COMPLEX OVERVIEW

The B-1B offensive and defensive avionics development program at Boeing is a classic example of a large and extremely complex avionics system. The avionic computers interface with a large number of LRUs and subsystems. Figure 1 shows a block diagram of the avionics system. At the center of the Boeing B-1B avionic system are six Avionic Control Units (ACUs) with six different operational flight programs in them. There are also two Data Transfer Units (DTUs) and a Mass Storage Unit (MSU) used for loading the operational software into the ACUs, loading mission data, and for recording flight parameters. The six ACUs and their major functions are as follows: GNACU - guidance and navigation; WDACU - weapon delivery; CDACU - controls and displays and defensive countermeasures; CRACU - critical backup to any of the three central ACUs, GNACU, WDACU, CDACU; TFACU - terrain following (primary and backup). The major means of communication is via MIL-STD-1553 buses with each ACU being a bus controller on at least one bus: GNACU - A bus; WDACU - B bus; CDACU - C bus; CRACU - D bus; TFACU - T

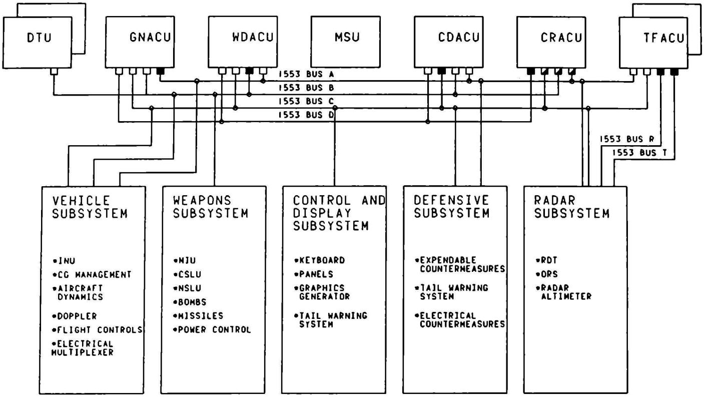


FIGURE 1 AVIONICS SYSTEM BLOCK DIAGRAM

and R buses.

In addition to these avionic computers and data storage units other avionic hardware/firmware units (LRUs) are integrated and tested with the rest of the system in the Boeing Systems Development Laboratory (SDL) as they become available. They include sensor conditioning interface units (the Radar Data Terminals - RDTs), weapons interface units (MIU, CSLU, NSLU), Inertial Navigation Units (INUs), and the Offensive Radar Subsystem (ORS).

In order to support the avionics development program the objectives of the B-1B simulation complex are as follows:

1) to provide a realistic environment for flight software development,
2) to provide the flexibility and accuracy required for comprehensive testing of flight software,
3) to provide a test bed for hardware /software integration, and
4) to provide a test bed for flight test problem resolution.

In order to meet these objectives, the following functions are required of the simulation complex:

- Vehicle System Simulation	(VSS)
- Weapon System Simulation	(WSS)
- Defensive System Simulation	(DSS)
- Radar Data Simulation	(RDS)
- Offline Test Software	(OTS)

Each of the above functions, except the Offline Test Software, resides in a separate Harris 800 computer system along with the simulation executive software. The OTS function is used before and after simulation runs and uses the vendor supplied operating system. Figure 2 shows a block diagram of the simulation complex. Four Harris computers make up a single "line". Each computer contains 256K words (24 bit) of memory and is connected to a 64K word shared memory unit. Peripherals used operationally include four general use CRT terminals, a single operator communication (OPCOM) terminal, a high resolution color graphics terminal, two 80 megabyte hard disk drives, and a 600 line per minute impact

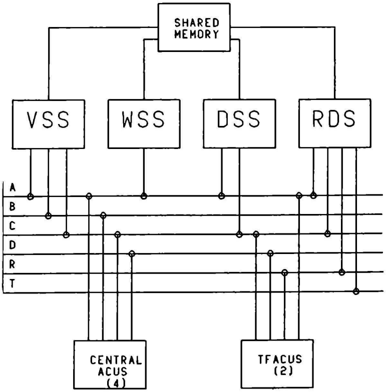


FIGURE 2 SIMULATION COMPLEX BLOCK DIAGRAM

printer. Each Harris also interfaces with Boeing built MIL-STD-1553 interface units and sensor interface units. Figure 3 shows the configuration of a single simulator computer.

In order to support different phases in the development of the avionics flight software and integration of avionic hardware, the simulation complex works in different levels and modes of simulation. Level I simulation refers to the case in which only the ACUs and their respective software loads are under test, i.e. all interfaces and LRUs are simulated by software models. Level II simulation refers to having selected real LRUs present. The simulation may also run in either a standalone or integrated mode. The standalone mode is when a single simulator is running on its own, interfacing with some portion of the flight software but not with the other simulators. All simulators except the WSS can run in the standalone mode. Integrated mode refers to multiple simulators operating together in a synchronized manner. Integrated mode has the VSS (master) running with any combination of the other simulators (WSS, DSS, RDS). An ACU synchronized mode is also possible in addition to levels I and II and integrated/ standalone. When ACU synchronized mode is selected, the simulator(s) are synchronized with the ACU major frame.

SIMULATION EXECUTIVE REQUIREMENTS

In order to support the simulation complex, the capabilities described in the following paragraphs are required of the real time executive.

SYNCHRONOUS SCHEDULING

The simulation executive is required to run synchronously with the flight software for two major reasons. First, it is necessary to have a common time reference (frame number) when recording data. This allows data recorded by more than one method (i.e. simulator or instrumentation device) to be correlated. Second, it is necessary to ensure the simulation model responses are within required limits. This is due to the fact that a status response may not be available for two frames after the command is sent. This maximum response latency can be guaranteed if the simulators are synchronized with the flight software in the ACUs (Figure 4a). The maximum response latency increases to three frames when the simulators and ACUs are not synchronized (Figure 4b). Without being able to guarantee a small maximum response latency time, the simulation models lose their effectiveness in simulating the real world.

The first item in Figure 4 illustrates the bus activity on the MIL-STD-1553 bus. The start of bus activity marks the start of the flight software frame. The next three items refer to simulator software processing. The first shows the initiation of the reading of 1553 data from the simulator 1553 interface unit. Since the interface unit buffers

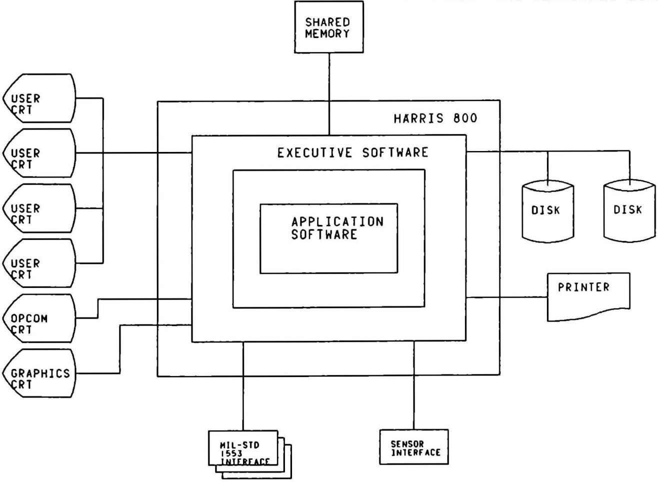


FIGURE 3 SIMULATION COMPUTER CONFIGURATION

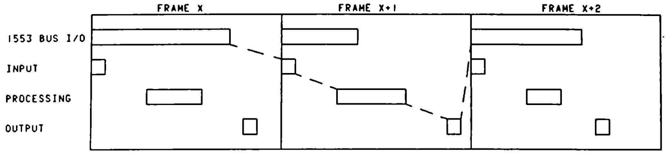


FIGURE 4A SIMULATOR RESPONSE LATENCY, SYNCHRONIZED WITH ACU

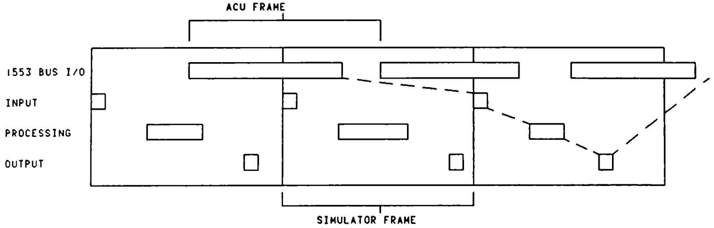


FIGURE 4B SIMULATOR RESPONSE LATENCY, UNSYNCHRONIZED

the 1553 data, the data being input represents bus data from the previous frame. When the data is read into the simulation, the models process and use the data. Any necessary responses are generated and the data is output to the 1553 interface unit. The response is then available for transmission to the flight software upon command.

ASYNCHRONOUS SCHEDULING

In addition to synchronous cyclic processing, it is necessary to have processing done on a cyclic basis which may be asynchronous to the flight software major frame. This is needed to provide a mechanism for processing both at a rate different than the flight software frame and processing which is not dependent on the flight software frame.

BACKGROUND SCHEDULING

The previous two scheduling types are referred to as foreground scheduling - scheduling which must be performed in order to keep the integrity of the simulation intact. Some simulation tasks are not as time critical as others and can thus be scheduled on a background basis, i.e. after all necessary foreground processing for a frame. These tasks still need to be performed but not in the strict manner of the foreground ones.

1553 BUS CONTROLLER/REMOTE TERMINAL

In order to support the simulation complex in its different operating modes and levels, the simulation executive has to be able to support both MIL-STD-1553 bus controller and remote terminal (RT) simulations. The bus controller simulation is necessary when a particular ACU is not integrated in the system and a simulator is needed to provide some of the ACU's bus commands and data to the ACUs and simulators that are present. The RT simulations account for most of the simulation modeling and a mechanism for simulating up to thirty one RTs per bus is required of the executive. There is only a single bus controller on each 1553 bus and it acts as the master while the remote terminals are the slaves.

DATA RECORDING/PRINTING

Given the complexity of the simulation complex and the avionics system and the amount of data being passed between the simulators and ACUs, a means for collecting data for later analysis is required. Both MIL-STD-1553 data and internal simulator parameters need to be recordable for later analysis in determining the success of a test. In order to assist the user in analysis during operation, the capability to process and print selected data is required.

FAULTING

One of the most important reasons for building simulation complexes is to have the ability to cause faults or off nominal conditions to occur and then to see whether the operational software responds correctly. For this reason, the simulation executive is required to provide a means to fault the outputs of the application software models.

CONTROLS AND DISPLAYS

In order to setup, monitor, and change simulation execution, controls and displays are required. The test conductor /user must be able to initialize the simulation configuration and models by selecting available options. Control is necessary during real time operation in order to turn on or off certain functions (i.e. data recording/printing and faulting) and to select data to be recorded or faulted. The user needs to be able to monitor the progress of the simulation throughout its operation. This causes a requirement for a display capability which allows many different items to be displayed to the operator. Due to limited amount of display devices (CRTs) and the number of items needed to be monitored at some time, a mechanism is needed to request different displays at any of the CRTs.

INTER-SIMULATOR COMMUNICATION

Inter-simulator communication is required in order to synchronize the simulators with each other and to provide data between the simulators. When the simulators run in an integrated mode, the VSS is always present with any combination of the other simulators. This is because the VSS is considered the "master" simulator in the simulation complex. It is the master because it supplies data to all the other simulators (WSS, DSS, RDS) and the other simulators synchronize to the VSS frame.

INITIALIZATION/LOADING

The simulators need the capability to load and initialize themselves. This is needed in order to get the software into memory, reset the data base, and reset the simulator hardware prior to use. This ensures a consistent and reliable starting point for simulator software execution.

DEVELOPMENT TOOLS

Software development tools and debugging aids built into the executive are required. Proper and efficient checkout, test, and regression testing is not possible without them. The ability to examine and modify any memory location as well as a cabability to step through and track software execution is necessary.

IMPLEMENTATION OF THE EXECUTIVE

Given the high level overview of the simulation complex and its functions and the requirements for the simulation executive, the following paragraphs describe the implementation of the executive developed.

The simulation executive is a standalone operating system which does not use the Harris operating system (VULCAN). This is due to the fact that there is a lot of inherent overhead when using a general purpose operating system and its device handlers. VULCAN also does not offer some capabilities required of the executive.

On the average, a couple of hundred context switches per second must be performed during simulation operation. A lot of time is used when VULCAN performs a context switch, so a more efficient scheduler was devised. The VULCAN device handlers also contain an enormous amount of overhead. Milli-seconds are used in validating an I/O request to a device under VULCAN in order to ensure proper authority for access to the device. These types of checks and restrictions are not required in the simulation so the overhead in I/O requests can be reduced to tens of microseconds by developing special simulation handlers.

Some capabilities required of the simulation executive can not be performed by VULCAN. The simulation software is required to be resident in memory at all times with the pages (1024 words) in consecutive order. VULCAN uses a virtual memory scheme in which only the active pages are kept in memory and not necessarily in any predictable order. This scheme does not support the timing requirements of the simulation so a unique loader needed to be developed in order to load and lock the simulator software into memory. In order to provide a useful C&D interface, the terminal interface needs to support full duplex. This allows operator inputs to be received while the displays on the terminal screens are being updated. The VULCAN terminal handler does not provide the capability, so a handler which does was developed.

In addition to the standard computer peripherals, the executive also interfaces with the Boeing built MIL-STD-1553 and sensor interface units. These units provide an interface between the simulation software and the flight software in the ACUs. In the case of the 1553 units, large amounts of data have to be managed and passed between the simulation computers and interface units.

Since the items just addressed comprise most of the pieces of an operating sys-

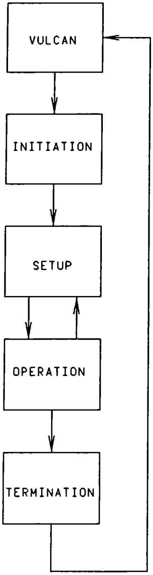


FIGURE 5 SIMULATION PHASES FLOW OF CONTROL

tem (scheduler, program loading and control, I/O handlers) the justification for the development of an entire operating system is valid.

PROCESSING PHASES

There are four phases in simulation processing: initiation, setup, operation; and termination. Figure 5 shows the flow of control between the different phases and the Harris operating system (VULCAN). The following paragraphs describe these phases in more detail.

INITIATION

The only interface between the executive and VULCAN is during initiation of the simulation software in which VULCAN services are used to assist in loading and assorted VULCAN data structures are copied to executive memory. Since the executive does not need VULCAN after loading and in order to utilize the available resources (i.e. memory) to the fullest, the executive loads itself over VULCAN. To do this the simulation software is loaded in stages (Figure 6). First, VULCAN is used to load the monitor executive loader from disk. This loader copies any data structures and data items needed from VULCAN memory to its own memory and then loads the monitor executive/user loader from disk. This first loader is only resident during the initiation phase and is written over during stage 3 loading. The monitor executive/user loader stays resident until the termination phase. It contains software used during the other phases and also loads the remainder of the simulation software. At the end of stage 3 the simulation software is the only software resident in memory and it controls CPU execution and the hardware interfaces. The executive resets all the hardware devices and decides if they are operational. It then resets its data base and schedules the application software for initialization processing. At the conclusion, the initiation phase is complete and the setup phase begins.

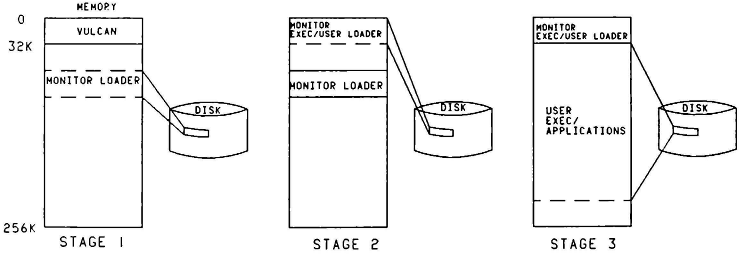


FIGURE 6 LOADING STAGES DURING INITIATION

SETUP

The setup phase is used to setup and configure the simulation models and functions. During the setup phase, only the executive software functions are active. The options of whether to run in a level 1 or level 2 configuration, to run integrated or standalone, or to run synchronously with the flight software are selected is this mode. The setup of the application models is also performed, such as the weapons selection for each bay. Other setups, which may also be performed in the operation phase are the specification of recording parameters, fault definitions, and the enabling of data recording and fault insertion. The setup phase always follows initiation and may also be reentered any number of times from the operation phase without having to terminate and reinitiate the simulation.

Part of the setup phase includes the transition to the operation phase. During the transition, the simulation executive and applications initialize according to the operator specified setups. Following this initialization, the executive waits for the specified synchronization source to start up before entering the operation phase. The synchronization source may either be the master simulator (integrated mode) or an ACU.

If simulator integration is selected, the executive sets up depending upon the application software (VSS, WSS, DSS, RDS) loaded. If the simulator is a VSS it is automatically integrated since the VSS is the master. If it is not a VSS, the executive sets up to synchronize itself with the executive in the VSS. This is accomplished by having the executive in the VSS write the current frame number in shared memory at the beginning of each frame and the non-VSS simulator executive monitor that location. The synchronization is done to the major frame (62.5 milliseconds). A major frame consists of four minor frames (15.625 milliseconds) and that is the rate to which the simulator interval timer is set during normal operations. When the timer expires, an interrupt is generated and an executive function is executed to handle the interrupt. In the VSS, every fourth interrupt (major frame) the major frame count in shared memory is incremented by the executive. In a non-VSS, the executive checks to see if the major frame count in shared memory has changed since the last time. If it has, the interval timer value is reduced, which in effect causes the next major frame to be shorter. This continues until the non-VSS executive sees no change in the major frame count between two checks. When this occurs, the executive continually monitors the shared memory frame count until either it changes or a time out occurs. If a time out occurs, the executive stops monitoring, resumes normal simulation execution and repeats checking with the next frame. This is necessary because the VSS may not be present and the non-VSS simulator would get stuck in the monitoring state. When a VSS is present and the non-VSS executive senses the major frame count change, the interval timer in the non-VSS is set faster than the VSS interval timer and the simulator starts executing synchronously with the VSS. The non-VSS minor frame is set slightly faster (15.6 vs. 15.625) than the VSS so that the non-VSS does not get behind the VSS execution. Since the clocks do drift slightly, the synchronization may be lost, in which case the non-VSS repeats the above procedure to get resychronized.

When ACU synchronization is selected, the executive also reacts according to the simulation configuration. If the simulators are running integrated, then only the VSS needs to synchronize with the flight software in the ACU's. This is because the other simulators are already synchronized with the VSS and thus will be synchronized with the ACUs when the VSS becomes synchronized. When the simulator is a VSS or a non-VSS running standalone, the executive must synchronize the simulator major frame with the flight software major frame. The ACU synchronization is handled differently than the simulator to simulator synchronization. For ACU synchronization, the real time clock (RTC) interrupt in the ACU is received by the simulator. The RTC in the ACU is set to trigger an interrupt every minor frame (15.625 ms) like the interval timer in the simulators. The executive also sets up to listen to the start frame message which is sent between the ACUs on the 1553 bus. This message is the first message sent on the bus every minor frame and consists of the running minor frame count. When the simulator is first synchronized to the ACUs the simulator minor frame count is set to the ACU minor frame count. This allows a common time reference to be established for use in recording data. While the simulator is synchronized with the ACU, the executive sets the simulator interval timer clock to run a little slow (15.7 vs 15.625 ms). It is reset to the slower value each time the RTC interrupt is received. In this way the interval timer clock acts as a watch dog timer. Under normal conditions it should never expire. If it does, it means the ACU to which it is being synchronized has failed in some way. When that happens the executive is able to continue by switching back to the interval timer for execution control.

OPERATION

Once the setup phase is completed and the simulator is integrated and/or ACU synchronized as requested, the simulation application models become active and start executing. Most of the processing is scheduled synchronously with the major frame (sixteen hertz) but the executive provides means for scheduling at other rates. Some functions need to execute at a rate of sixty four hertz (15.625 ms) such as 1553 bus controller processing.

The executive also provides a means to schedule a task or tasks at other rates. All scheduling is controlled by task blocks. These task blocks are used to signal the scheduler to run a task. The ordering of the blocks defines the priority and types (foreground/background) of each task. A task may be triggered from a clock source (interval timer or real time clock) or due to the occurrence of some event or condition as defined by the requesting software.

TERMINATION

The termination phase is entered upon completion of the operation phase due to operator request or fatal error condition. The executive is responsible for bringing the simulation to an orderly conclusion. All external interfaces are disabled and the final recording buffers are written to disk. If a fatal error occurred, the appropriate information is written to the OPCOM terminal prior to initiating the reload of VULCAN. In addition, the contents of simulator memory are written to disk for use in analysis of the error. The reload of VULCAN is performed at the conclusion of all terminations.

PROGRAM CONTROL

The simulation executive consists of tasks, control functions, and service functions. Figure 7 shows the flow of control between the executive function and simulation tasks (executive and application) during setup and operation. Some of the tasks are strictly executive (disk services, printer processing, command processing) or application (foreground B, application foreground and background) or are a combination of both (foreground A, sixty four hertz, 1553 input). The executive control and service functions are used for all the tasks.

CONTROL

The executive control function consists of the multitasking scheduler and the clock interrupt handlers. It controls the execution of the simulation tasks and the synchronization of the simulator with another simulator or ACU. The interval timer clock, simulator real time clock, and ACU real time clock all trigger the control function for task execution. In addition, tasks may trigger

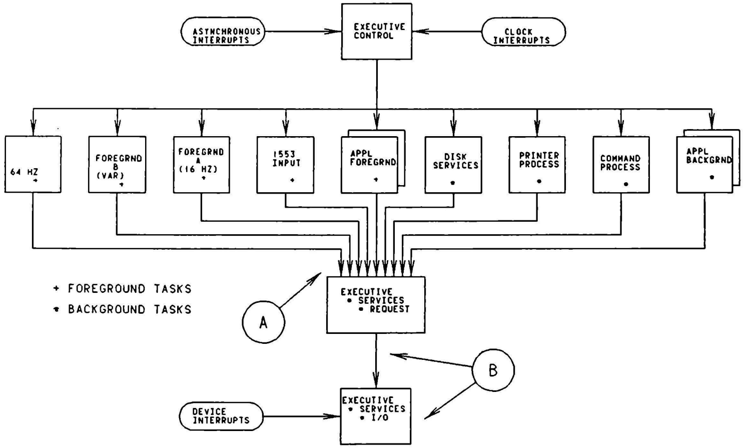


FIGURE 7 SIMULATION TASKS/EXECUTIVE FLOW OF CONTROL

the control function as required for asynchronous task execution. The scheduler checks the task blocks for the highest priority requested task each time it is triggered. Foreground tasks are checked first, followed by background tasks. The control function allows foreground processing overflow of one entire minor frame (15.625 ms) before aborting the simulation. The initial synchronization and any resynchronization due to clock drifts are handled by the executive control function.

The control function also provides some debug capabilities. These consist of an address trap and breakpoint functions. The address trap is a hardware/software feature which monitors a specified memory location and saves the CPU registers, program counter, and memory contents each time the memory location is either modified or executed. The breakpoint function provides a means to step through simulation execution in a non-real time mode.

SERVICES

The executive services function provides the interface between the simulation software and hardware. This includes the CRT's, disk drives, printer, MIL-STD-1553 interface units and sensor interface units. The services function is divided into request and I/O subfunctions. The request function accepts requests and processes them. The I/O function initiates reads/writes and services the device interrupts.

Data recording is handled by providing the means to specify tables defining the recording parameters. For internal simulator data this is done by using tables of internal addresses to be recorded. MIL-STD-1553 data recording is done by use of tables of specific words in specified messages.

There are two types of internal data recording, interval and event triggered. Up to one hundred simulator addresses may be defined for the interval recording table. An interval (in major frames) is also specified. When enabled, a snapshot of each of the memory locations specified in the table is taken at the specified interval and the data is stored in a recording buffer. When the buffer is full it is written to a dedicated disk file. There are two event triggered recording tables. Each can contain up to sixteen addresses and an address to monitor (once a frame). When the contents of the monitored address changes, the snapshot is performed and the data is buffered. There are two disk files dedicated for these snapshots.

1553 data recording is selectable by 1553 bus. One RT on each bus is dedicated by the flight software as the data pump address. When enabled, data going to the data pump address is brought into the simulator, buffered up and written to a dedicated disk file.

1553 data faulting is used to introduce faults in the 1553 data to the flight software from the simulation models. The faulting is done non-destructively. That is, the "good" data is always available for the simulation application software but a faulted data buffer for each message is maintained. When a fault is enabled, the good data value is either replaced by an entire word (analog fault) or the data value is altered by applying a set and reset mask to it (discrete fault). The faulted data is then output to the interface unit for transmission to the flight software.

Due to the multitasking capability of the executive, the executive services software must protect itself from reentrancy problems. The "A" in Figure 7 identifies the problem. Since any of the tasks can be using an executive request service and since that task can be context switched-out for a higher priority task to run, the executive request services must either be reentrant (i.e. do not modify any local data or code) or inhibit context switching. Both methods of protection are implemented.

A similiar problem exists for the executive I/O services software. "B" in Figure 7 identifies the concern. Here, the problem occurs because the I/O services software can execute upon a call from the request services software or from an I/O device interrupt. The only way this contention is handled is to disable the particular interrupt(s) when critical portions of code have to be executed.

TASKS

The simulation tasks are divided into executive only, application only, and executive/application tasks. Figure 7 shows the simulation tasks.

A simulator may have a sixty four hertz task. The executive bus controller function executes at sixty four hertz. Foreground A processing is scheduled at a sixteen hertz rate. Most of the simulation processing is done at this rate. A portion of the controls and displays software executes in this task. Foreground B provides the application software with a variable rate cyclic processing task. The 1553 input task executes when new 1553 data has been received from the interface units. The 1553 messages are validated by the executive and dispatched to applications for processing. In addition, other foreground application tasks are possible. These tasks must be triggered by another task when appropriate.

The background tasks only execute when time is available in the current frame. The disk services task is requested and runs when a disk file is either being deleted, created, or expanded.

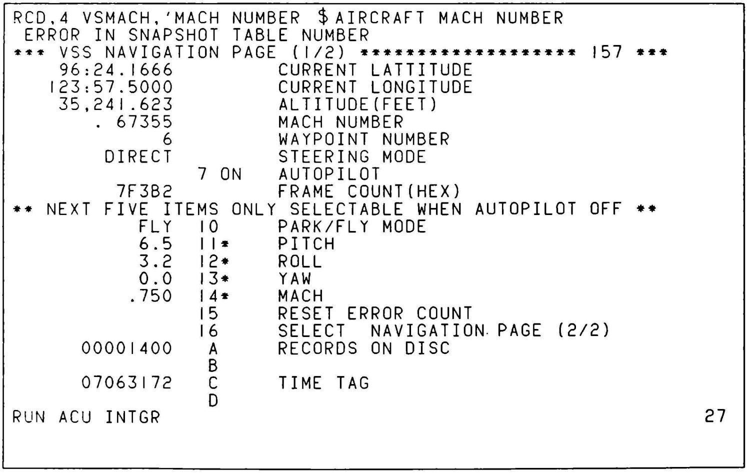
The printer task is used to process print requests. Due to the inherent slowness of the printer and the desire to not lose print data during operation, a spool/despool technique is used. If print requests are being made slow enough (less than 600 lines per minute), the request is processed immediately. When the requests are made too quickly, they are written to disk and read back in as the printer becomes available.

The command processing task handles the parsing and processing of the coded commands from the operator. Single commands are entered either from the user CRTs or from an operator specified disk file.

USER INTERFACE

The operator interfaces with the simulator via the controls and displays (C&D) function. The interface is both menu and command driven. Up to four terminals are provided for the C&D interface. A fifth terminal is used for display of diagnostic and error messages. The format of the displays on the four terminals is the same and is shown in Figure 8. The top line is the command input and data insertion line. Coded commands are entered by the operator for such things as definition of recording and display parameters and simulation control. The second line is used for echoing the commands and inserting data values or displaying an error message when a bad command or data item is entered. The next portion of the screen represents the "C&D page". It is the portion of the screen over which other pages (menus) are placed when selected.

Each page can contain up to sixteen items. An item can be a switch (on/off) (item 7), data insertion (items 11 thru 14), page select (item 16), button (items 10 and 15), or display only (items 1 thru 6 and 8). In order to select an item, the corresponding function key (f1 = item 1) on the keyboard is selected by the operator. A switch toggles the represented memory location between two values as defined in the C&D page definition. A data insertion item allows the contents of a memory location to be altered. Upon selection, the new data value (hexadecimal, octal, logical, decimal integer, floating point, or angular) is keyed in by the operator. A button item causes a special function in the simulator to be performed. This could be anything from the resetting of a variable to the starting of a series of computations. Display only items are identified by the lack of an item number present on the page. The operator has no control over the display. Page select items cause another C&D page to be written to the screen. The C&D pages



```
RCD,4 VSMACH,'MACH NUMBER $AIRCRAFT MACH NUMBER
 ERROR IN SNAPSHOT TABLE NUMBER
*** VSS NAVIGATION PAGE (1/2) ******************* 157 ***
    96:24.1666          CURRENT LATTITUDE
   123:57.5000          CURRENT LONGITUDE
    35,241.623          ALTITUDE(FEET)
        . 67355         MACH NUMBER
             6          WAYPOINT NUMBER
        DIRECT          STEERING MODE
                  7 ON  AUTOPILOT
         7F3B2          FRAME COUNT(HEX)
** NEXT FIVE ITEMS ONLY SELECTABLE WHEN AUTOPILOT OFF **
           FLY    10    PARK/FLY MODE
           6.5    11*   PITCH
           3.2    12*   ROLL
           0.0    13*   YAW
          .750    14*   MACH
                  15    RESET ERROR COUNT
                  16    SELECT  NAVIGATION PAGE (2/2)
      00001400    A     RECORDS ON DISC
                  B
      07063172    C     TIME TAG
                  D
RUN ACU INTGR                                          27
```

FIGURE 8 SAMPLE CONTROL AND DISPLAY PAGE

are organized in a heirarchical fashion. Each page may have a father, sibling, and children pages. Special keys (^, <, >) are used to select father and sibling pages while children pages are selected using page select items. Each page has a unique number and may also be accessed by use of a coded command.

The next portion of the screen represents the memory display area. Up to four memory locations may be displayed at a given terminal. The locations displayed are operator selectable via coded command. The last area is the status line. It shows the current state and status of the simulation.

One additional feature of the C&D function is the playback mode of operation. During normal operation every operator input (item selection, data insertion, command input, key hit) is recorded to a disk file along with the relative time of its input. This file can then be used as input during a subsequent simulator run. When playback mode is selected, all operator interaction is disabled until completion. During the suspend (setup) mode of the simulation, the inputs are processed in an accelerated fashion and in the run (operational) mode, the inputs are processed at the same relative times as originally entered. These inputs are recorded again during the playback mode which makes additions to the saved setup possible. This feature consistently allows fast, efficient, and error free setup of the simulator.

CONCLUSIONS

The real time executive developed for use in the Boeing B-1B avionics simulation complex met or exceeded all of its requirements. It has proved to be an extremely powerful and flexible package of software. The capabilities it provides cover all current and foreseeable requirements or changes to the simulation software. It took approximately ninety man months of effort over a one and a half year period to develop the executive. As a result, the efficiency of the simulation executive exceeded initial estimates. For example, the disk I/O handler proved to be twenty five percent faster and the terminal handler fifty percent faster than VULCAN's. The executive utilizes twenty four percent of the major frame (15 out of 62.5 ms) during peak usage and thirty two percent of available memory (82K out of 256K).

As avionic systems continue to grow in size and complexity, the need for complex and large simulations will follow. Just as the real time executive described here was the "heart" of the simulation complex, so the future simulations will require an executive just as powerful. While the executive was built to control the B-1B simulation complex, it can be used as the basis and starting point for other simulation executives. The capabilities it provides are broad and general enough to be able to apply to a wide range of simulation applications both in single and multiple computer simulation complexes. The design of the executive is well structured and documented in great detail. Documented naming conventions and programming standards were followed in order to aid in the maintainability and transportability of the real time executive.

ACKNOWLEDGEMENTS

The author would like to acknowledge the following people for their contributions to the executive:

Ron Pehrson for leading the development and designing a large portion of the executive.

Jerry Stanton for a major part of the executive design and his unique insight and solutions to problems during the development.

85-1756

AIRCRAFT TACTICAL ENVIRONMENT SIMULATION FOR THE 1990'S

S. Ramachandran*
D. K. Oldham**
R. L. Branson†

Goodyear Aerospace Corporation
Akron, Ohio

Abstract

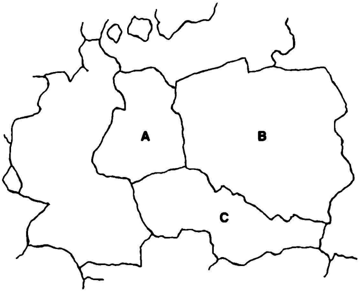
The requirements of aircraft tactical environment simulation are addressed in this paper in light of current and anticipated developments. The problems and issues involved in providing realistic pilot training in a dense threat environment and a laboratory simulation to investigate these issues are discussed.

Future Tactical Environment

The tactical aircrew will face a forbidding threat environment in future conflicts. The accelerating arms buildup of our potential adversaries has led to a very high concentration of threats that our pilots will face even within a single mission area. The pilot will be confronted by a wide variety of threats such as fighters, helicopters, surface-to-air missiles (SAM), and anti-aircraft artillery (AAA). His task is made all the more difficult by the numerous enemy radars, jammers, and other electronic warfare (EW) systems that will attempt to increase his vulnerability while frustrating his mission objectives. A better appreciation of the threat situation for simulation can be gained from the analysis below.

Figure 1 shows the current deployment of Northern Warsaw Pact forces[1, 2]. The number of active threats in a gaming area is estimated to be 1470, and the number of active threats in a 1600 square mile target area to be 147 for this deployment. Data from References 1, 2, and 3 were used, and the following assumptions were made to obtain this estimate: 1) a tactical gaming area of 160,000 square miles, 2) a factor of 0.1 for gaming vs. target area distribution of forces, and 3) 50% active forces. For a total tactical environment simulation, these numbers will be much higher when radars, jammers, reconnaissance aircraft, helos, and other elements that constitute the total threat environment are included. It must be noted that this analysis is based on current national deployment in these three countries but does not include any Soviet forces that are deployed there now. Hence, threat density could be considerably higher in the 1990's based on the arms build-up and the range and performance of new weapon systems cited in Reference 4. For these reasons, it may be reasonable to expect about 2000 active threat elements in a gaming area and about 200 active threat elements in a target area. The tactical pilot in an air defense role faces equally awesome numbers. According to Reference 4, an air attack on key NATO airfields may involve 300 to 400 aircraft, and an assault in the Central NATO Region could involve 2400 aircraft.

There is another dimension to the tactical environment that will make the situation even more difficult for pilots. Advances in technology, both planned and in development, will significantly expand the sensor and EW capabilities from radio frequency (RF) into electro-optical (EO) and infrared (IR) parts of the electromagnetic spectrum. The complexity and diversity of the tactical environment simulation is graphically shown in Figure 2. When interaction among the numerous elements is taken into account, it becomes obvious that the problem is even more complex than that illustrated.



A

- 359 COMBAT A/C
- 200 SAM LAUNCHERS AT 30 SITES
- 96 AA MOBILE GUNS

B

- 675 COMBAT A/C
- 400 SAM LAUNCHERS AT 50 SITES
- 880 AA MOBILE GUNS

C

- 439 COMBAT A/C
- 250 SAM LAUNCHERS AT 40 SITES
- 600 AA MOBILE GUNS

Figure 1. Northern Warsaw Pact (Non-Soviet) Threat Concentration[1, 2]

Role of Simulators

Aircrew survival and mission success are highly dependent upon the ability to provide realistic training. This training must transfer skills to successfully counter and

*Staff Engineer, Member AIAA
**Engineering Specialist
†Senior Development Engineer

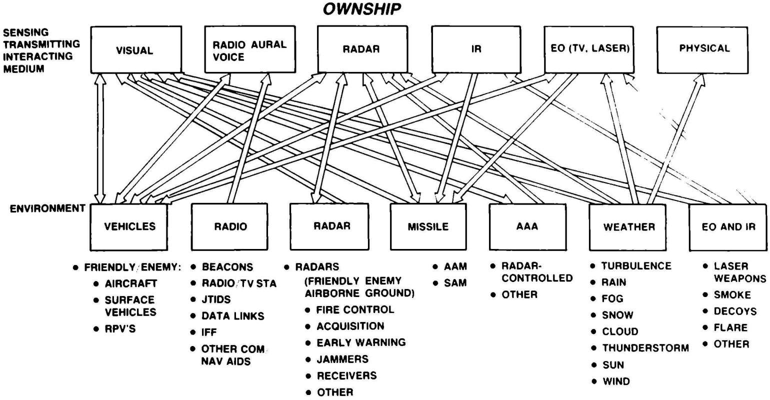


Figure 2. Typical Aircraft Tactical Environment

neutralize threats through effective use of tactics, countermeasures, and counter-counter-measures. The training must enable the crew to apply the skills even while flying through a very dense environment that could contain thousands of threat elements. To quote General Gabriel, "You must train the way you will fight," and "realism is the key."[5] It is not practical to train the aircrew in the real world against an actual hostile environment, but there are two alternatives to provide this training. One method is to build training ranges and perform training such as the red flag exercises. While this method could enhance realism, there are several disadvantages to this approach. It is very expensive to build a threat environment containing hundreds of threat elements and keep the hardware updated as potential adversaries field new systems or upgrade existing equipment. Such ranges cannot be easily used for mission training since they lack the geographical and feature exactness of the mission area. Also, the absence of weapon fire will considerably reduce the psychological perception of threat intensity.

The other method is through flight simulators. A flight simulator is an ideal medium to impart such training and complement red flag type exercises. The safety and cost benefits derived from simulator use are obvious. Often it yields other unexpected benefits too. For example, a recent USAF investigation with a limited field-of-view visual for the F-15 simulator found that while the visual system substantially enhanced the pilot combat training as expected, it also provided two unanticipated potential capabilities.[6, 7] One was an improvement in the pilot's decision-making ability about when to eject in emergency situations. The second was the mission rehearsal training provided by the simulator.

Simulation Requirements

To provide a positive and effective training in the simulator will require a complete and realistic simulation of the tactical environment including a high degree of interaction between ownship, mission supporting forces, and the numerous threat elements as well as interaction amongst the threat elements themselves. Current training simulators provide neither such a dense, high interactive environment nor the needed realism. Their environment is limited to one-on-one, or at most, one-on-two combat situations. Table 1 shows the threat simulation capability of some of the USAF simulators. Their emphasis was on EW. It must be observed that the limitation of these simulators is primarily due to the time frame in which they were designed when the predicted threats were not numerous and computer technology was not advanced enough to provide real-time simulation of a large number of threats.

There are several other important factors that affect the training capability of a simulator but are not apparent from Figure 2. A brief discussion of some of these follows.

Command, Control and Communications (C^3) Simulation: While much initiative and free actions are taken at the unit level in the real world, the entire offensive and defensive forces act in a cohesive manner to obtain maximum effectiveness. Enemy war tactics and actions propagate through a C^3 network affecting the way a mission proceeds. For example, effects of elec-

Table 1. Number of Simulated Threats* in USAF Trainers[8]

Trainer	Mission Maximum	Maximum Instantaneous
F-4	9	9
F-5	30	10
A-10	100	20
F-111	16	16
F-15	61	15

*Jammers, Artillery, Radars, Missiles, and Platforms tronic counter measures (ECM) and electronic counter-counter-measures (ECCM) determine how and where an intercept will take place. They also determine what resources will be available to the enemy for engaging an attacking aircraft. Hence, a reasonably extensive C^3 simulation is a very important part of the tactical environment.

Simulation vs Stimulation: Whether to stimulate ownship onboard equipment such as central computers, signal processors, etc., or to simulate them depends very much upon the specific aircraft that is simulated. For example, the RF environment will be by design much less transparent to the pilot of a single seat fighter than to the crew of an electronic support measures (ESM) aircraft. In the former case, there may be no training value or need for detailed spectral displays and threat parameter readouts; thus, a functional simulation may suffice. Some other factors that influence selection of simulation or stimulation are: complexity of algorithms executed in the processors, length of program code, and computational power. The main drawback to the stimulation approach is that it demands much more hardware than simulation. This includes actual aircraft systems that usually present acquisition difficulties. Further, specially designed signal generators and interface hardware are also needed. John Lethert's paper contains a good discussion on some other aspects which affect the simulation vs. stimulation decision.[8]

Instructional System: The advent of a dense threat environment in a training simulator will cast an additional burden on the instructor. This can detract him from his principal function of training an aircrew. The task of controlling and monitoring 2000 plus threat elements, keeping track of pilot actions, and generating an audit trail for debrief can simply overwhelm an instructor during a training session. Instructional aids will need to be developed to lighten his load and to help him set up the problem and monitor the progress of the mission. They should also permit him to interact and make changes to the problem as he sees fit, evaluate and score the pilot, and provide an effective debriefing. Emerging technologies such as artificial intelligence (AI)/expert systems could provide a solution to this problem as well as to the C^3 simulation.

Traditionally threat simulation has emphasized the electronic warfare aspect of the environment concentrating on the radar aspect of the threats. It is obvious from the preceding discussions that the environment simulaton must be all-encompassing to provide the necessary intensity and realism in a threat environment and to offer pilot challenge. A unified approach is therefore essential to simulate the diverse elements of the tactical environment and to achieve correlation between the various sensors, visual system, and electronic warfare (EW) simulation.

Computational Considerations

The role of tactical environment simulation as we saw earlier is to generate all the information on the outside world as required by the ownship and its sensors. Its functions are diverse and extensive. These include supplying enemy aircraft attitude and position information to the simulator visual system, providing RF signal strength of an active search radar to the EW simulation, furnishing IR signature information of an enemy air target to the weapon system simulation, providing the wind turbulence components to the aerodynamics and radar simulation, and simulating jammed communications, to name a few. Considering the all-encompassing nature of the environment simulation, it is important to look at the computational resources that will be required to create the level of threat saturation discussed earlier. To assess this, an analysis of computer software for most of the elements of Figure 2 was performed. Existing computer codes from various simulation projects were used to provide a rough estimate for the worst case analysis with FORTRAN as the programming language. It was assumed that there will be a total of 2000 active elements in the mission gaming area, and 200 of them will be instantaneously interactive. It was found that the worst case computational load will correspond to all the threats being smart targets[9] that are updated at a host computer sampling rate of 30 Hz. (Smart targets can maneuver adaptively against enemy aircraft as in a real-world air combat.) The computer power required for this scenario will be 8.0 MFLOPS.

This is a tremendous amount of computer power just for the environment simulation. Creative approaches will be required to make the environment simulation viable and affordable. One envisioned solution is the segmented approach to simulation shown in Figure 3. The overall simulation is broken down into various segments based on computational requirements and environment density. As an example, an airplane engaged in a dogfight with the ownship or an enemy missile within kill range to the ownship will need to be computed at a fast rate (say 30 Hz) as a top priority threat, enemy aircraft outside the ownship maneuvering range will be simulated at a moderate rate (say 10 Hz) in the instantaneously interactive mode, and a search radar looking for an intruder will be simulated at a slow rate (say 1 Hz) as a continuously active element. The C^3 itself can be simulated at a still lower rate (say 0.5 Hz).

Range, closing rate, and nature of threat could be used as discriminators for segment allocation. With this approach, the total number of elements in any of these segments decreases as computational intensity increases. For example, it will be physically impossible for the ownship to simultaneously engage all of the 200 enemy aircraft in a dogfight, and only a small number of them need to be simulated at 30 Hz while the rest can be relegated to the instantaneously interactive mode that

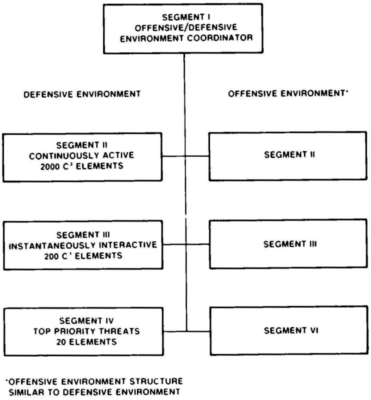


Figure 3. Example of Segmented Approach

runs at a slower rate. Preliminary analysis shows that this approach could reduce computational requirements by a factor of five to ten.

The rapid progress in computer technology holds the promise that the simulation can be done at a modest cost in the near future. The concept of concurrent processing[10, 11] may make it possible to realize a solution with just a few microprocessors since the environmental simulation is ideally suited for parallel processing. Some possible hardware configurations presently under consideration by Goodyear include Goodyear's Massively Parallel Processor technology, Goodyear's Multipurpose Integrated Processing System (MIPS-301) and Intel's iPSC series computers.[12]

Goodyear Laboratory Simulation

Goodyear Aerospace Corporation (GAC) has set up a laboratory simulation to investigate in depth the problems, issues, and approaches that were discussed above and to arrive at practical solutions for implementation of a comprehensive and realistic real-time tactical environment simulation for aircrew training.

A system block diagram of the lab simulaton is shown in Figure 4. The simulation consists of a Gould 32/67 computer (host) with peripheral equipment (three 300-mb discs, one 80-mb disc, tape drives, line printer, and a few Televideo terminals), special purpose hardware (an array processor and several microcomputers) for high-speed computation, two ADAGE displays, a joystick, a set of function switches, and a graphics tablet. One ADAGE display serves as the pilot display and the other as the instructor display. The pilot display provides airspeed, altitude, attitude, and heading information via a simulated head-up display (HUD).

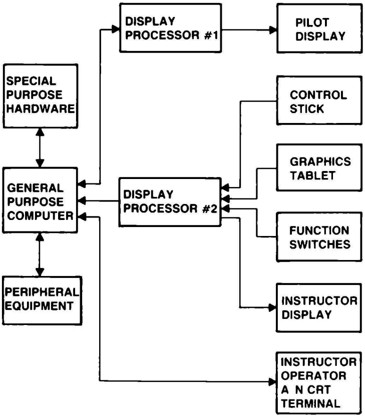


Figure 4. Laboratory Simulation System Block Diagram

The pilot display also features a representative EW display and provides information on active enemy threats. It shows threat type, mode, bearing and range relative to ownship. A radar map as seen by the ownship can be displayed but is yet to be integrated into the simulation.

The instructor display shows the gaming area land mass in color. Symbols for ownship, targets, SAM sites, AAA sites, radar sites, and enemy aircraft are superimposed on the land mass. Threat status and missile trajectories are provided to the instructor giving him a complete plan view of the mission status.

Ownship is flown by the pilot using the joystick, and the ECM is activated through function switches. The instructor controls and monitors the problem through the A/N terminal and function switches. The graphics tablet is used for mission scenario setup and positioning of threats and targets on the land mass.

The investigation will be carried out in phases starting with the RF & IR environment currently under development (see Figure 5). EO, communications, and other elements of Figure 2 will be added at appropriate stages. In Figure 5, the offensive forces are shown as 'blue' and the defensive forces as 'red.' The blue aircraft consist of one ownship (flown by the human pilot in the loop), and the rest are computer-controlled and typically a mix of fighters, stand-off jammers, early warning aircraft, etc. The red aircraft are all computer-controlled and may have a mix of aircraft as appropriate to the defensive role. The defensive forces contain other threat elements also such as various types of radars, missiles, etc.

All computer-controlled fighters are smart aircraft. The blue and red forces are controlled and coordinated by individual Command and Control (C^2) structures suitable for the forces they represent. Additionally, the environment simulation includes various types of radars, jammers, SAM's, etc., as required, and other supporting components as shown in Figure 5.

Presently, the lab simulation has the capability to fly the ownship, modeled after the F-15, against multiple red aircraft and various types of red SAM's and AAM's in an interactive fashion. The pilot and instructor displays are fully operational. The rest of the simulation shown in Figure 5 is under development. The findings of this investigation will be reported in a future forum.

Concluding Remarks

The tactical pilot may encounter well over a thousand threats during a typical mission in future conflicts. The simulator is an ideal medium to train aircrew and to ensure pilot survivial and mission success in dense threat environments. The tactical environment is very diverse and complex, and a unified approach to its simulation is essential to cover all aspects and to achieve realism and correlation among various simulated subsystems. A laboratory simulation is under development to investigate and solve this problem.

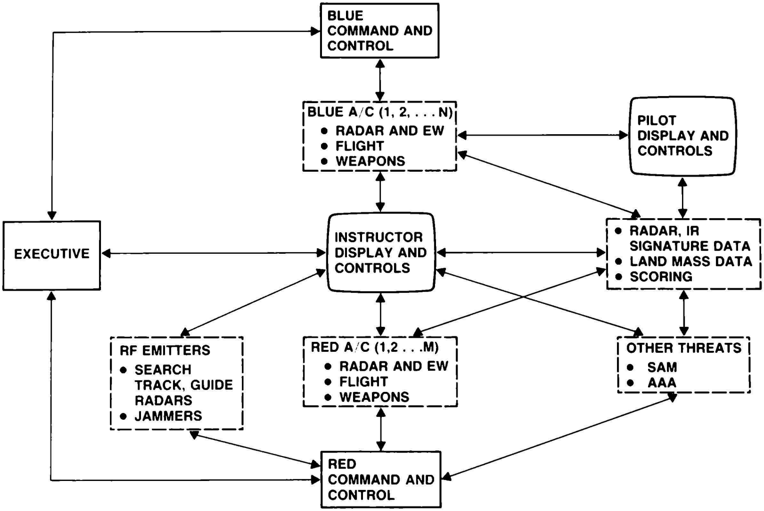


Figure 5. Phase I Tactical Environment Simulation

References

1. "The Warsaw Pact," Air Force Magazine, December 1984, pp. 96-98.

2. "The East-West Conventional Balance in Europe," Air Force Magazine, December 1984, pp. 156-157.

3. Taylor, John W. R., ed., Jane's All the World Aircraft 1984-1985, Jane's Publishing Company Ltd., London, England, 1984.

4. "U.S. Air Forces in Europe - Soviets Deploy Updated MiG-25 Foxbat Fighter," Aviation Week and Space Technology, June 7, 1982, pp. 54-56.

5. Gabriel, Charles A., Gen. USAF, Chief of Staff, "Tactical Air Reinforcement for Europe," NATO's Sixteen Nations, Special Issue 2, Vol. 29, No. 4, 1984, p. 91.

6. O'Neal, Maston E., Lt. Col., USAF, "F-15 Limited Field of View Visual Training Effectiveness Evaluation," TAC Project 83G-066T, USAF Tactical Air Warfare Center, July 1984.

7. Coyne, James P., "Aerospace World - News, Views and Comments," Air Force Magazine, April 1985, pp. 38-39.

8. Lethert, John F., "Electronic Warfare Simulation for Air Force Weapon System Trainers," AIAA Paper 78-1590.

9. Tobler, Sheldon A., "Development of a Simulation of a Smart Target for Air Combat," GER 17045, Goodyear Aerospace Corporation, Dec. 1981.

10. Seitz, Charles A., "The Cosmic Cube," Communications of ACM, Vol. 28, No. 1, Jan. 1985, pp. 22-23.

11. Fox, G. C. and Otto, Steve W., "Algorithms for Concurrent Processors," Physics Today, May 1984, pp. 13-20.

12. "Intel's Personal Super Computer (iPSC)," Sales Brochure, Intel Corporation, Beaver, Oregon, 1985.

ROTOR DYNAMICS SIMULATION: A NEW APPROACH

85-1739

Frank M. Cardullo*
State University of New York
Binghamton, New York

William J. Hewitt
Consultants in Simulation, Inc.
Harpursville, New York

Katie Knight
AAI Corporation
Houston, Texas

Abstract

Currently the two most popular rotor dynamics simulation techniques are the "blade element" method and the "rotor map" method. The "blade element" method calculates the lift, drag and pitching moment of small blade elements at a specific number of radial stations and a specific number of azimuthal stations. The lift, drag, and pitching moment of these elements are then summed over the entire rotor disk to determine rotor performance. The "rotor map" method employs the storage of the rotor aerodynamic coefficients C_T, C_Q, C_Y, C_m, C_1,

as functions of the three independent variables μ, λ and $\theta_{.75}$. Both of these

methods have serious deficienies. The method discussed in the paper attempts to combine the two aforementioned techniques in a manner which retains the strength of each while mitigating their relative weaknesses. This technique which is referred to as the Performance Driven Blade Element Method makes the blade airfoil aerodynamic coefficients C_L and C_D four dimensional functions of λ, μ, θ,

and element angle of attack. These functional coefficients will be stored in data maps suitable for linear interpolation between entries. This then makes thousands of data entries available for adjustment to facilitate performance correlation. To further enhance this method, an optimization technique is presented which systematically adjusts the functional coefficients to correlate with aircraft performance for each performance test point. The optimization algorithm minimizes the cost functional, which may be performance error, by adjusting the functional coefficients associated with the test point in question.

Introduction

The real time simulation of helicopter rotor dynamics is considered by many to be an "open problem". This perception is held because there are significant compromises associated with the two most frequently used state-of-the-art methods, the "blade element" method and the "rotor map" method. While the former method is purported to yield excellent dynamic fidelity, it has severe limitations in meeting stringent tolerances on steady state performance over the entire flight regime. The latter method has the inherent flexibility which allows efficient tailoring to accurately meet steady state performance. However, this method does not provide the same high dynamic fidelity as the "blade element" method.

Background

In order to better understand the method presented in this paper, it is useful to briefly discuss the above introduced methods. The "blade element" method calculates the lift, drag and pitching moment of small blade elements at a specific number of radial stations and a specific number of azimuthal stations. The lift, drag and pitching moment of these elements are integrated over the entire rotor disc to determine the overall force and moment components acting on the rotor disc. These forces and moments are then summed with the other forces and moments acting on the helicopter and are subsequently used in the equations of motion to compute the helicopter state vector. All major degrees of freedom; the six rigid degrees of freedom of the fuselage, flapping of the rotor, rotor angular velocity, longitudinal cyclic and lateral cyclic are automatically accounted for in this approach.

As stated previously, a shortcoming of the "blade element" method is that it does not readily meet static performance criteria over the entire flight regime. Aircraft performance includes speed/power cruise performance at various weights, altitudes and stores configurations over the entire speed regime. In addition, hover performance in and out of ground effect must be satisfied at various weights and altitudes. The simulator must also meet climb and descent performance standards as well as altitude ceilings. These performances must usually be satisfied within a tolerance on the order of five percent.

The kernal of the "blade element" method consists of the two dimensional lift and drag coefficients of the blade airfoil section at the nth blade element. These coefficients may take the form of analytic functions of angle of attack as:

*Member AIAA

$$C_{L_n} = a_0 + a_1\alpha_n + a_2\alpha_n^2 \qquad (1)$$

$$C_{D_n} = \delta_0 + \delta_1\alpha_n + \delta_2\alpha_n^2 \qquad (2)$$

In order to adjust rotor performance, the six coefficients in the above equations are all that are readily available to be manipulated. These do not offer sufficient flexibility to satisfy the aircraft performance data over the entire flight envelope.

The "rotor map" method employes the storage of the rotor aerodynamic coefficients C_T, C_Q, C_Y, C_m, C_l as functions of the three independent variables μ, λ, $\theta_{.75}$ which are cross flow ratio, inflow ratio and blade pitch at the 0.75R station. These then determine the steady state rotor performance with a high degree of accuracy. Further, these tables include thousands of coefficient entries which may be adjusted to assure compliance with performance criteria.

As can be seen from the previous discussions, the above two methods are somewhat complementary, the strength of one is the weakness of the other. Hence, if the two methods could be somehow combined such that their relative strengths are maximized and their weaknesses minimized, an optimum approach may be realized. This is the basis of the "performance driven blade element method".

Description of the Method

The "Performance Driven Blade Element Method" (PDBEM) was developed to satisfy two major criteria. The first criterion is to meet steady state performance standards over the entire envelope. The second is to provide satisfactory dynamic performance. The manner in which this is accomplished is to employ an off line model to automatically compute the rotor performance at each required point and then to use an optimization algorithm to systematically adjust the blade element functional coefficient data until the performance criteria are satisfied. Then, the resulting coefficient data are stored in tables to be used by a blade element model in real time to compute the rotor forces and moments. Figure 1 illustrates the method.

In more detail, the PDBEM makes the airfoil aerodynamic coefficients C_L, C_D three dimensional functions of μ, λ, $\theta_{.75}$ and blade-element angle of attack (α_R). For example, recalling equations (1) and (2);

$$a_i = f_i(\mu, \lambda, \theta_{.75}); \text{ for } i = 0, 1, 2 \qquad (3)$$

$$\delta_i = f_j(\mu, \lambda, \theta_{.75}); \text{ for } j = 3, 4, 5 \qquad (4)$$

These functional coefficients (f_i and f_j) will be stored for table lookup with linear interpolation between entries. The result of this is to have thousands of points in a data table available for adjusting to facilitate performance correlation. Initially, the tables will contain only the values of the a_i and δ_j. However, as the process of tailoring the data proceeds, these constants will be altered.

Each performance point of interest is associated with a semi-unique set of μ, λ, and $\theta_{.75}$. In addition, there are semi-unique table entries in the vicinity of each semi-unique set.

As was stated previously, the methods involve executing a performance model off line to determine if the coefficients currently stored in the tables will provide steady state performance within the ascribed tolerance and, if they don't, employing an optimizing algorithm for the purpose of adjusting the data until the tolerances are met. In order to get an idea of the magnitude of this effort, an estimate of the number of steady state performance test points was made. In the operations manual for the aircraft, there are on the order of 300 steady state performance test points. Automatic execution of these 300 test points is generally part of the software capability of current simulators. Therefore, nothing special needs to be done here to implement this facet of the PDBEM.

The optimization routine (OPTIMO) can employ many optimization algorithms to adjust

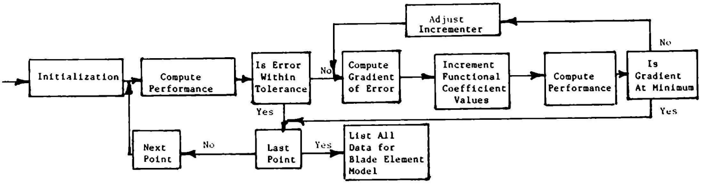


Figure 1

the functional coefficients in the tables. For the purposes of demonstrating the technique for this paper, a steepest descent iterative algorithm was employed to minimize the cost functional, which was chosen to be the square of performance error, by adjusting the functional coefficients associated with the test point in question. The optimization technique selected here is well suited for this problem because steepest descent yields rapid convergence and it employs the gradient which incorporates each functional coefficient (f_i) in its definition. The convergence criterion implemented herein is that when the gradient is a minimum the cost function is a minimum.

Results

A simple performance model was developed for the purpose of demonstrating the method. This model uses, as the performance parameter, rotor torque. In this case, the lift and drag coefficients are computed from;

$$C_{L_n} = \eta_R(f_1 a_0 + f_2 a_1 \alpha_R) \qquad (5)$$

$$C_{D_n} = f_3 f_{(MT)} + f_4 \delta_1 \alpha_R + f_5 \delta_2 \alpha_R^2 \qquad (6)$$

where:

η_R is the recirculation efficiency

α_R is the blade element angle of attack

$f_{(MT)}$ is the trim Mach No. correction to the blade element minimum drag coefficient.

Then the in-plane drag force is computed from;

$$D_{IP} = TF(-C_L \sin\Delta\alpha_R + C_D \cos\Delta\alpha_R) \qquad (7)$$

where: TF is thrust factor

$\Delta\alpha_R$ is the blade element angle of attack when $\theta_{.75} = 0$.

Finally, the torque is computed as being a function of the in-plane drag force;

$$R_{Q_m} = D_{IP}\,(.75R) \qquad (8)$$

The demonstration was executed for 30 performance test points including hover, level cruise and vertical climb cases. One set of data was generated using actual performance results. Subsequently a second set was generated which yielded some error in torque. These results and their associated coefficient data were then used as inputs to the optimization algorithm. The optimizer then systematically adjusts the tabulated coefficient values (f_1 ......f_n) until the error in torque is reduced to within the prescribed tolerance.

The optimization algorithm reported on herein is a single point optimizer which adjusts each performance point separately. A multiple point optimization scheme has also been developed.

Conclusion

A new method for simulating rotor dynamics was presented. It was stated that this method is based on the principle of minimizing the steady state performance error by automatically systematically adjusting the functional coefficients in an off-line environment. The resulting functional coefficients are subsequently used in a real time blade element model which computes the rotor forces and moments used in the aircraft equations of motion. The presented method has the inherent advantages of matching steady state performance as achieved by rotor map methods and providing the dynamic capability of the blade element approach.

References

1. Briczinski, S.J. "The Concept of the Rotor Map Method for Modelling Helicopter Rotor Characteristics and its Application to Trainer Type Simulators," Presented at FAA Helicopter Simulator Certification Meeting, Atlanta, Georgia. April 18, 1984.

2. Ferguson, S.W. "A Mathematical Model for Real Time Flight Simulation of a Generic Tilt-Rotor Aircraft," NASA CR166536, October 1983.

3. Howe, R.M. "Real Time Helicopter Simulation using the AD-10," Applied Dynamics Publication.